ОЗ – Инструкция за учителя: Съдържание

Сборник от работни листове за учителя, базирани на дейности за проектиране на игри





Информация за документа

Резултат по проекта: О3 / А1 – Сборник от работни листове за учителя, базирани на дейности за проектиране на игри
Интелектуален резултат № - ОЗ: Заглавие – Инструкция за учителя
Ръководител: Югозападен университет "Неофит Рилски" (България)

Участвали партньори: Университет в Любляна (Словения), Университет в Риека (Хърватия)

Автори:

Отговорност

Този проект е финансиран от програма "Еразъм +" на Европейския съюз.

Информацията и вижданията, изложени в тази публикация, са на автора (авторите) и не отразяват непременно официалното мнение на Европейския съюз. Нито институциите и органите на Европейския съюз, нито каквото и да е лице, действащо от тяхно име, не могат да бъдат държани отговорни за използването на съдържащата се в тях информация.

Coding4Girls, 2018-2020



Creative Commons Attribution-ShareAlike 4.0 International Public License (<u>CC BY-SA 4.0</u>)





Съдържание Основни учебни сценарии 8 Учебен сценарий 1 - Въведение в Snap! интерфейс 8 Учебен сценарий 2 - Време е да оживите вашия спрайт 13 Учебен сценарий 3 - Придвижване по сцената 17 Учебен сценарий 4 - Смяна на костюми и обръщане 23 Учебен сценарий 5 - Звуци от фермата 29 Учебен сценарий 6 – Лятната ваканция на хамелеона 35 Учебен сценарий 7 – Помогнете на принца и принцесата да открият техните 45 животни Учебен сценарий 8 – Рисуване с креда (тебишир) 52 Учебен сценарий 9 - Прибиране на боклука и почистване на парка 63 Учебен сценарий 10 - Хранене на котките 71 Учебен сценарий 11 - Познай броя на котките в приют 78 86 Сценарии за напреднали Учебен сценарий 12 – Улов на здравословна храна 86 Учебен сценарий 13 – Разказване на история 95 Учебен сценарий 14 – Рисуване 105 Учебен сценарий 15 – Хвани мишката 113 122 Учебен сценарий 16 – Купуване на храна за пикник Учебен сценарий 17 – Операции 130 Учебен сценарий 18 – Рециклиране 136 Учебен сценарий 19.2 – Свири на пиано 147 Учебен сценарий 20 - Тест 156 Учебен сценарий 21 - Опростена игра РАСМАN 161 168

Литература





увод

Водещи психолози от миналия век определят играта като една от най-важните дейности за развитието на важни житейски умения, независимо от възрастта или етапа на развитие. Детето чрез играта бързо се приспособява към нови обстоятелства и се справя с промяната с лекота. Когато играе, той открива основни понятия от истинската дума и се правят първите основни връзки между тях.

В днешно време игрите се използват по-често в най-ранните етапи от развитието на детето у дома и в детската градина. Обучението в училище все още твърде често се основава на традиционно предаване на знания в рамките на ориентиран към учителя модел с пасивни ученици. От друга страна, теориите за обучение, разработени през миналия век, насърчават нови подходи към преподаването и ученето, които са ориентирани към учениците, базирани на проблеми, насочени към по-високо познавателно равнище на образователните цели (на по-високи таксономични нива), мотивационни и често подкрепяни от ИКТ.

Подходът CODING4GIRLS ще насърчи участието в дейности по програмиране чрез "подход с висок таван с нисък вход", който в началото има ниски изисквания за знания, като същевременно не ограничава предизвикателствата за решаване на проблеми за по-напредналите учещи. Учащите ще бъдат насърчавани да довършат частично завършени решения чрез добавяне на липсващи градивни елементи на кода или да създадат свои собствени решения. Дейностите се планират последователно, от основни със само едно понятие за програмиране до по-напреднали с множество понятия и процедури за програмиране. Докато подготвяхме учебни дейности в Snap !, ние се фокусирахме върху идентифицираните характеристики на игрите, предпочитани от момичетата, и върху дейностите, свързани с реалните проблеми.

Подготвените учебни листове със сценарии, представят в съкратен формат информация, която ще помогне на учителите да интегрират предложените сериозни игри и методологии за учене чрез подхода на дизайн мисленето в техните преподавателски практики. Те следват CODING4GIRLS подхода, основан на активно учене, дизайн милене и проектиране на обучението и включват информация за всяка учебна дейност, която трябва да се разработи за изграждане на умения за програмиране за момичета и момчета.

Налична е следната информация:

- Обща образователна цел на съответния урок;
- Понятия, обхванати от урока;
- Специфични учебни цели;
- Очаквани резултати от обучението;
- Използване стъпка по стъпка на подхода за обучение CODING4GIRLS, основан на игрово базирано обучение и дизайн мислене;
- Методи за оценка на усвоените знания;
- Въпроси за иницииране на дискусия между учащите се в контекста на съвместни дейности в клас и работа в екип.





Изготвени са 22 учебни листа, съответстващи на учебните дейности. Учителите могат да използват сценариите и игрите в предложената последователност или могат да ги избират свободно според своите предпочитания и нужди. Учебните листове обхващат както общата функционалност на предложената сериозна игра, включително процеси на взаимодействие с потребителите и генериране на обратна връзка, така и описания на всички учебни дейности, които ще бъдат приложени в предложената сериозна игра. Учебните листове са на разположение на английски, както и на националните езици на партньорите по проекта - български, хърватски, гръцки, италиански, португалски, словенски и турски.





Учебни листове

Подготвените учебни листове варират от основни, с въвеждане и затвърждаване на една концепция за програмиране, до по-напреднали с използване на множество концепции за програмиране. Следващата таблица представя предложения ред на дейностите.

Основ	зни обучителни сценарии	
1	Въведение в Snap! интерфейс	
1	Запознаване със Snap! среда за визуално блоковопрограмиране	UL
	Време е да оживите вашия спрайт	
2	Намерете блокове за програмиране, свържете ги, преместете спрайта,	UL
	накарайте спрайта да каже нещо.	
	Придвижване по сцената	
3	Създаване на смислена последователност от блокове за движение на	UL
	героите (спрайтовете).	
4	Смяна на костюми и обръщане	UL
F	Звуци от фермата	
5	Добавяне, импортиране, запис и възпроизвеждане на звук.	UL
	Лятната ваканция на Хамелеона, опростена версия	
6	Запознаване със събития, чувствителност на цветовете, логически	UL
	стойности, проверка и реагиране на две различни състояния на играта	
7	Помогнете на принца и принцесата да открият техните животни	
/	Използване на условни, рисуване	UL
Q	Рисуване с креда (тебешир)	111
0	Използване на цикли/loop/, обръщане, промяна на фона (сцената)	UL
	Прибиране на боклука и почистване на парка	
9	Запознаване с променливи, дублиране на спрайтове, части/blocks/ от	UL
	код	
	Хранене на котките	
10	Използване на променливи (вътре / извън цикъла), цикли,	UL
	произволни числа, конкатенация на низове, оператори, вход.	
	Познай броя на котките в приют	
11	Използване на случайни числа, въвеждане на променливи, условни	UL
	блокове, оператори за сравнение, брояч.	
ADVA	NCED LEARNING SCENARIOS	
	Улов на здравословна храна	
12	Използване на променливи, условни, цикъл, точка в посока,	UL
	произволни	
13	Разказване на приказки	SWU
14	Рисуване	UNIRI
15	Хвани мишката	UI
12	Използване на цикли, условни условия, променливи	
16	Купуване на храна за пикник	111
	Използване на променливи, условни условия, оператори	01





17	Операции Използване на случайни числа, смяна декор на сцена, вход, условни блокове	SWU
18	Рециклиране	SWU
19.1	Свири на пиано 1	SWU
19.2	Свири на пиано 2	UNIRI
20	Тест	SWU
21	Опростена игра РАСМАN Използване на движение на обект въз основа на събития, сензори на цветовете, логически стойности, проверка и реагиране на две различни състояния на играта	UL





ОСНОВНИ УЧЕБНИ СЦЕНАРИИ

Учебен сценарий 1 - Въведение в Snap! интерфейс

Учебен сценарий	Въведение в Snap! интерфейс						
Име							
Предишен опит в	Не се изисква						
програмирането							
Очаквани резултати	Общи резултати от обучението:						
	• Запознаване със Snap! среда за визуално блоково						
	програмиране						
	конкретни резултати от обучението:						
	• ученикът може да добави нов спрайт						
	 ученикы може да дооави костюм кым спраит и да то редактира 						
	• ученикът е в състояние да центрира спрайта, така че						
	въртенето да работи правилно						
	 ученикът може да дооави нов фон за сцената и да го редактира 						
Цел, задачи и	Ученикът добавя нов спрайт, добавя костюм към спрайта,						
кратко описание на	редактира костюма и изтрива един от тях. Ученикът създава нов						
дейностите	фон на сцената, редактира го и изтрива нежеланите.						
	Цел: До края на часа учениците ще нарисуват любимия си герой						
	и жизнената му среда, реална или въображаема, за да го						
	използват в игра. В научните изследвания рисуването на						
	спрайтиве е единтефицирано като подходящо за целевата						
	група. Това прави дейността по-мотивираща за всички ученици.						
Продължителност	45 минути						
Стратегия и методи	Учителят демонстрира дейността						
на обучение	Самостоятелна работа						
Форми на обучение	Работа с класа/групата (фронтални дейности)						
	Индивидуална работа						
Резюме на	(Мотивация-въведение, прилагане, осмисляне и оценка)						
обучението	В края на часа учениците ще нарисуват любимия си герой и						





неговата среда на живот, реална или въображаема, за да го използват в бъдеща игра.

[Стъпка 1]

Покажете на учениците уеб страницата, където могат да намерят Snap! (<u>https://snap.berkeley.edu/</u>).

Покажете им различни части на интерфейса: раздел с блокове, раздел, където те могат да сглобяват скриптове, да сменят костюми, да добавят звуци, сцена със спрайт върху него, списък на спрайтове.





Можете да създадете нов спрайт, като щракнете върху един от

трите бутона:







Ще се опитате да нарисувате нов спрайт, затова щракнете върху четката и ще се отвори изскачащ прозорец, където можете да нарисувате вашия спрайт по подобен начин като в Paint. Задача за ученици: Създайте първия си спрайт. Имате 10 минути. След изготвянето на спрайта трябва да се уверите, че центърът на въртене на спрайта е там, където искате да бъде. Задача за ученици: центрирайте вашия спрайт [Стъпка 3] За да редактирате вашия спрайт, изберете раздела Костюми, който се вижда само когато щракнете върху вашия спрайт. Щракнете с десния бутон върху костюма, който искате да редактирате, и изберете редактиране. Можете също да дублирате костюма си или да го изтриете със същото меню. untitled ⇔ 1 Sprite(2) 🗸 draggable Costumes Scripts Sounds Ê. edit rename duplicate delete export [Стъпка 4] За да импортирате вече съществуващ костюм, щракнете върху иконата с нарисуван лист хартия и изберете Костюми ...





Co-funded by the Erasmus+ Programme of the European Union



Отново, тази опция ще се покаже само когато вашият спрайт бъде щракнат под сцената.

<u>Задача за учениците:</u> изберете костюм и го добавете към спрайта [Стъпка 5]

Сега имате своя герой и трябва да добавите малко фон към сцената. За да направите това, първо щракнете върху сцената вместо върху героя под сцената. За да добавите нов фон, изберете раздела Фонове:



Задача за учениците: нарисувайте своя фон.

<u>Задача за учениците:</u> претърсете съществуващите фонове и добавете един от тях, така че да имате два.

<u>Задача за учениците</u>: Намерете начин да редактирате фона си. Намерете начин да изтриете един от вашите фонове, така че да остане само един.





	Рефлексия и оценка:
	Успяли ли са учениците да нарисуват своя характер и обкръжение
	там, където живее? Имали ли са проблеми? Как ги решиха?
Инструменти и	https://snap.berkeley.edu/
ресурси за учителя	
Инструменти и	Инструкции за учениците (C4G1_InstructionsForStudent_BG.docx)
ресурси за	
учениците	





Учебен сценарий 2 - Време е да оживите вашия спрайт

Учебен сценарий	Време е да оживите вашия спрайт						
Име							
Предишен опит в	/						
програмирането							
Очаквани резултати	Общи резултати от обучението:						
	 Ученикът знае къде да намери определени блокове за програмиране и как да ги свърже в последователност Ученикът знае как да премества спрайт Ученикът знае как да накара спрайта да каже нещо Специфични резултати от обучението, ориентирани към алгоритмично мислене: Създаване на смислена последователност от блокове 						
Цел, задачи и	Ученикът открива къде се съхраняват блоковете за програмиране						
кратко описание на	и как да намери подходящите, какви категории блокове има и как						
дейностите	да свърже блоковете в последователност						
Продължителност	45 минути						
Стратегия и методи	Учителят демонстрира дейността						
на обучение	Самостоятелна работа						
Форми на обучение	Работа пред класа/групата						
	Самостоятелна работа						
Резюме на	(Мотивация-въведение, прилагане, осмисляне и оценка)						
обучението	Ще накарате героя си да се движи и да каже нещо през този час.						
	Можете да им покажете пример за програма, която те ще						
	програмират през този час.						
	[Стъпка 1]						
	Първо нека да разгледаме къде са налични програмните блокове,						
	които можете да използвате. Къде са те?						
	От лявата страна можете да намерите различни категории						
	блокове: Motion, Looks, Sounds, Pen, Control, Sensing, Operations и						
	Variables. Първо ще използваме блокове move O steps .						
	Задача за ученици: Първо намерете блока и след това щракнете						





двукратно върху него. Какво се случва?
[Стъпка 2]
За да започнете да свързвате блоковете в програма, трябва да
плъзнете и пуснете блока точе стрет в раздела Скриптове.
Image: Sound Control Sensing Sprite Sound Operators Image: Sound steps Scripts Costumes Sound steps
turn 👌 (15) degrees 🕂 👘 move (10) steps
Можете да щракнете двукратно върху блока в раздела Скриптове,
за да изпълните кода.
[Стъпка 3]
Програмите в Snap! обикновено започват с щракване върху
зеления флаг.
Задача за учениците: щракнете върху различни типове категории
и се опитайте да намерите блок, който стартира програмата, ако
се щракне върху зеления флаг.
Решение:
Motion Control Looks Sensing Sound Operators Pen Variables Scripts Costumes when clicked move 10 steps when space key pressed when 1 am clicked
Ако искате програмата да работи в правилна последователност от
стъпки, блоковете трябва да бъдат свързани както при пъзелите.







каже нещо. Направете повече от едно движение. Изпробвайте.





Завърща ли се персонажът на точно същото положение всеки път,				
когато програмата ви се изпълнява? Можете ли да намерите блок,				
който да гарантира, че вашият герой ще започва винаги от една и				
съща позиция и няма да излиза извън сцената?				
Съвет за учителя: ако героят "избяга" от сцената, можете да го				
извикате обратно на сцената, като щракнете върху него с десния				
бутон на мишката и изберете show.				
Блокът, който търсите, е 📴 to x: 🔿 y: O . За да определите кои				
стойности за х и у са подходящи, можете да преместите героя си				
на мястото, на което искате да бъде, и щракнете върху х позиция				
и позиция у (в долната част на блоковете в категория Motion) и				
текущите х и у ще се покажат. Просто трябва да ги запишете в				
белите полета в go to блок.				
Рефлексия и оценка:				
Колко пъти вашият герой трябваше да повтори кодът за движение				
и да изпълни последователсността "кажи", за да изпълни				
задачата? Еднакво ли е числото за всички в класа? Защо е така?				
Примерна програма:				
https://snap.berkeley.edu/snap/snap.html#present:Username=spelac				
<u>&ProjectName=C4G dog goes home</u>				
• Инструкции за ученика(C4G2_InstructionsForStudent_BG.docx)				
• Ако ученикът не е нарисувал собствен спрайт и фон, той може				
да използва:				
 <u>https://snap.berkeley.edu/snap/snap.html#present:Username=s</u> 				
pelac&ProjectName=C4G dog goes home tmp				





Учебен сценарий 3 - Придвижване по сцената

Учебен сценарий	Придвижване по сцената						
Име							
Предишен опит в	Ученикът знае къде да намери блокове за програмиране и как да						
програмирането	ги свърже в последователност						
Очаквани резултати	Общи резултати от обучението:						
	• Създаване на смислена последователност от блокове						
	Специфични резултати от обучението, ориентирани към						
	алгоритмично мислене:						
	• Ученикът позиционира спрайта на сцената						
	• Ученикът променя х и у позицията на спрайта						
	 Ученикът използва цикъл repeat n 						
	• Ученикът научава, че посоката на движение на спрайта в						
	move steps е спрямо посоката, към която е обърнат						
	спрайтът.						
Цел, задачи и	Кратко описание: Ученикът се учи как да премести своя спрайт в						
кратко описание на	позиции х и у на сцената, програмира лесна програма за						
дейностите	решаване на задачите, научава се как да завърти спрайта си в						
	различна посока и как това се отразява на блока move steps.						
	Задачи: Създайте програма, която премества спрайт по позиция х,						
	създайте програма, която премества спрайт в позиция у, създайте						
	програма, която комбинира движение в позиции х и у.						
	Цел: Ученикът прави разлика между движението в позиции х и						
	у на сцената и използва цикъл.						
Продължителност	45 минути						
Стратегия и методи	Учителят демонстрира дейностите пред класа/групата						
на обучение	Самостоятелна работа						
Форми на обучение	Фронатална работа/ групова работа						
	Самостоятелна работа						
Резюме на	(Мотивация-въведение, прилагане, рефлексия и оценка)						





обучението	Ще помогнете на различни животни да постигнат целите си. За						
	целта ще трябва да им дадете инструкции как да се движат по						
	сцената.						
	[Стъпка 1]						
	Отворете <i>Catch the ball</i> и добавете код на кучето, така че то да						
	хване топката. Използвайте блоковете сhange x by 🔘 и						
	wait O secs за да направите анимация на куче, което се движи						
	към топката.						
	Възможно решение на задачата:						
	when by clicked						
	go to x: (-150) y: (-80)						
	wait 1 secs						
	change x by 20						
	wait 1 secs						
	wait 1 secs						
	change x by 20						
	wait 1 secs						
	change x by 20						
	wait 1 secs						
	change x by 20 wait 1 secs						
	change x by 20						
	wait 1 secs						
	change x by 20						
	change x by 20						
	wait 1 secs						
	change x by 20						
	wait 1 secs						
	change x by 20						
	change x by 20						
	wait 🕦 secs						
	change x by 20						
	change x by 20						





Както се вижда, х се променя, когато се движите наляво или
надясно. Ако х е 0, вашият спрайт е в средата на сцената. Всичко,
което е вляво от центъра на сцената изисква пред числото за х да
поставите знак минус "-". Колкото героят е по-далече от центъра,
толкова, по-голямо е числото за позиция х. Вдясно от средата,
стойностите на х са числа, по-големи от 0.
Съвет: Ако се направи с по-големи ученици, които познават
десетичните дроби, времето за изчакване може да бъде по-
кратко, напр. 0,1. Ако те знаят какво е координатна система,
някои обяснения могат да бъдат пропуснати или описани с
понятийния апарат за координати и координатна истема.
[Стъпка 2]
Отворете Помогни на маймуната да се изкачи на дървото (Ір
monkey climb the tree) и добавете код към маймуната, за да
донесе бананите.
Използвайте блокове <mark>Сћапде у Бу О</mark> и <mark>Wait Osecs</mark> ,за да
направите анимация на маймуна, която се катери на палмата.
Възможно решение на задачата:





Co-funded by the Erasmus+ Programme of the European Union

vhen Na clicked ga ta x: 🚺 y: 💶 change y by 🕕 ait (1) s change y by 📶 it 🕦 secs change y by (10) vait 1 secs change y by (10) wait 1 secs change y by 🔟 vait 🕦 secs change y by 📶 ait 🕦 s change y by 🔟 rait 🚺 secs change y by 1 vait 🚺 secs change y by 🕕 wait 1 secs change y by (10) wait 🕦 secs change y by 10

Както се вижда, у се променя, когато се движите нагоре или надолу. Ако у е 0, вашият спрайт е в средата на сцената. Всичко, което е по-високо от средата, има у по-голямо от 0. Ако искате вашият спрайт да бъде под средната линия на сцената, то е точно като да се гмуркате: казвате, че сте под водата, като поставите знака минус - отпред на числото и кажете, колко "метра" под водата сте и на сцената казвате - колко стъпала под средната линия сте. Ако искате да се спуснете обратно от дървото, използвайте change y by c10.

[Стъпка 3]

И в двете стъпки трябваше да използвате взаимозаменяемо два блока. Колко пъти трябваше да повторите кода? Има по-кратък начин за писане на този код, като кажете на

20





компютъра да повтори вашия код определен брой пъти. Това е
блок за цикъл repeat Можете да го използвате, когато едно и
също действие или последователност от действия се повтарят
повече от веднъж. Опитайте се да промените кода си и за двете
repeat 🔵
задачи, така че да използвате 🖵 Кодът, който искате да
повторите, трябва да бъде поставен вътре в този блок и трябва да
напишете колко пъти трябва да се повтори в празното
пространство.
Код за кучето:
when clicked go to x: -150 y: -80 repeat 13 wait 1 secs change x by 20
Код за маймуната:
when clicked go to x: 0 y: 120 repeat 11 wait 1 secs change y by 10
Задача: Опитайте се да накарате кучето да тича към топката и
обратно.
Задача: Опитайте се да накарате маймуната да се изкачи на
дървото и да се върне надолу.
Какво ви хареса най-много? Можете да си помогнете с х и у
позицията на спрайта, като използвате XY Grid background в Snap:





			Y	(X:0,Y:180)		
			0			
			2.00			
	(X:-24	0,Y:0)		(X:0,Y:0)	(X:240	, Y:0)
	,200	,200		200	200	X
			200			
				(X:0,Y:-180)		
Инструменти и	•	Възможно решение за Catch the ball:				
ресурси за учителя		https://snap.berkeley.edu/snap/snap.html#present:Username				
		<pre>=spelac&ProjectName=C4G moving x</pre>				
	•	Възможно решение за Help monkey climb a tree:				
		https://snap.berkeley.edu/snap/snap.html#present:Username				
		<pre>=spelac&ProjectName=C4G moving y</pre>				
Инструменти и	•	Catch the ball (Хвани топката):				
ресурси за		https://snap.berkeley.edu/snap/snap.html#present:Username				
учениците		<pre>=spelac&ProjectName=C4G Catch the ball</pre>				
	•	Help monkey	climb the tree	(Помогни на и	маймуната да с	e
		изкачи на дървото):				
		https://snap.t	perkeley.edu/s	nap/snap.html	#present:Usern	<u>name</u>
		<pre>=spelac&ProjectName=C4G Help monkey climb the tree</pre>				
	•	Инструкции за ученика				
		(C4G3_InstructionsForStudent_BG.docx)				





Учебен сценарий 4 - Смяна на костюми и обръщане

Учебен сценарий	Смяна на костюми и обръщане
Име	
Предишен опит в	Движение
програмирането	
Очаквани резултати	Общи резултати от обучението:
	• Създаване на смислена последователност от блокове
	Специфични резултати от обучението, ориентирани към
	алгоритмично мислене:
	• Ученикът променя костюма на спрайт, за да направи
	анимация
	• Ученикът променя завъртането на героя (спрайта).
Цел, задачи и	Кратко описание: Ученикът се учи как да промени костюма на
кратко описание на	спрайта, за да направи анимация. Той също така се научава как да
дейностите	променя различните видове завъртане на спрайта.
	Задача: Създайте програма, която променя костюма на спрайта;
	във всяка програма задайте подходящ тип завъртане за всеки
	спрайт
	Цел: Да се усвои начинът за смяна на костюма на спрайта и как да
	се зададе подходящ тип завъртане на спрайт.
Продължителност	45 минути
Стратегия и методи	Демонстрация пред класа/групата
на обучение	Самостоятелна работа
Форми на обучение	Презентация пред класа/групата
	Самостоятелна работа











създадете код, който ще започне, когато се щракне зеления флаг, и 15 пъти променя всяка секунда променя външния вид на Балерина. Ще трябва да използвате блок **next costume**. Уверете се, че вашата Балерина започва и завършва танца си с двата крака на пода. Началната и крайната позиция не са част от нейния танц. Решение: when 🍋 clicked switch to costume ballerina'a repeat (15) next costume wait 1 secs switch to costume ballerina'a [Стъпка 2] Нашата балерина не иска да бъде постоянно на една и съща позиция, затова прави малки движения всеки път, когато сменя костюма. Добавете това движение към нейния танц. Възможно решение: when 🍋 clicked switch to costume ballerina'a repeat (15) next costume move (10) steps wait 🚹 secs switch to costume ballerina's [Стъпка 3] Отворете нов празен проект и импортирайте всички "ходещи" костюми на Awery (Ейвъри). Добавете подходящ фон, по който да върви Ейвъри. Създайте анимация на Ейвъри, ходеща от лявата страна на сцената до дясната страна на сцената. Опитайте се да разберете как да анимирате Ейвъри по начин, че нейните стъпки да изглеждат свързани, както в реалния живот. Възможно решение:

















	се върти така, както искате, трябва да щракнете върху съответнат		
	икона отляво на вашия спрайт:		
	e d d	Mouse ✓ draggable cripts Costumes Sounds	
	Кръгл	ата стрелка означава, че вашият спрайт може да се обръща	
	във вс	яка посока (като мишката).	
	Стрелката <-> означава, че спрайтът ви ще се обръща само наляв		
	или надясно (това е, което бихте използвали кучето да не ходи		
	"на главата си")		
	Последната -> стрелка означава, че спрайтът винаги ще изглежда		
	такъв,	какъвто е (можете да използвате това за маймуната)	
	Опита	йте се да пренапишете кодовете си за кучето и маймуната,	
	така ч	е те първо да преминат обекта и да се върнат обратно, като	
	се обт	ърнат. Уверете се, че сте променили правилно стила им на	
	върте	He.	
Инструменти и	•	Програмни решения за балерина:	
ресурси за учителя		https://snap.berkeley.edu/snap/snap.html#present:Username	
		=spelac&ProjectName=C4G dancing	
	•	Avery ходене:	
		https://snap.berkeley.edu/snap/snap.html#present:Username	
		<pre>=spelac&ProjectName=C4G Avery walking</pre>	
	•	Решение за намиране на сиренето:	
		https://snap.berkeley.edu/snap/snap.html#present:Username	
		<pre>=spelac&ProjectName=C4G Find cheese solution</pre>	
Инструменти и	•	Намиране на сиренето:	
ресурси за		https://snap.berkeley.edu/snap/snap.html#present:Username	
учениците		=spelac&ProjectName=C4G Find cheese	
	•	Инструкции за ученика	
		(C4G4_InstructionsForStudent_BG.docx)	





Учебен сценарий 5 - Звуци от фермата

Учебен сценарий	Звуци от фермата	
Име		
Предишен опит в	 Ученикът може да добави фон. 	
програмирането	 Ученикът може да добави нов спрайт. 	
	 Ученикът знае как да накара спрайта да каже нещо. 	
Очаквани резултати	Общи резултати от обучението:	
	 добавяне звук от медийната библиотека на Snap, 	
	 импортиране на звук от други носители, 	
	• запис на нов звук,	
	• выпроизвеждане на звук при напискане на клавиш. Специфични резултати от обучението ориентирани към	
	алгоритмично мислене:	
	• ученикът добавя звук от медийната библиотека на Snap и	
	го възпроизвежда при натискане на определен клавиш,	
	• ученикът импортира звук от компютър и го възпроизвежда	
	при натискане на определен клавиш,	
	• ученикът записва нов звук и го възпроизвежда при	
Пел залачи и	натискане на определен клавиш. Кратко описание: Програмирайте проста игра в която играчът	
цел, задачи и		
кратко описание на	научава звуците на животни, като натиска определени клавиши.	
деиностите	задачи: в първата стъпка ученикът тряова да изоере фона на	
	сцената. След това трябва да програмира жената фермер да казва	
	инструкциите:	
	1) Ако искате да чуете кучето, щракнете върху бутона "D" !;	
	2) Ако искате да чуете кравата, щракнете върху бутона "С" !;	
	3) Ако искате да чуете овцете, щракнете върху бутона "S" !;	
	4) Ако искате да чуете прасето, щракнете върху бутона "Р" !;	
	5) Ако искате да чуете коня, щракнете върху бутона "Н" !.	
	След това ученикът трябва да програмира задачата според	
	указанията на жената фермер.	
	Цел: Учениците ще бъдат запознати как да добавят нов звук и как	
	да го използват. Те също така ще се научат как да използват	
	звуковия блок ("възпроизвеждане на звук [name_of_sound]") и	
	контролния блок ("при натискане на бутона [When the the_key is	





	pressed]").		
Продължителност	45 минути		
Стратегия и методи	Активно учене; обучение, основано на проектиране на		
на обучение	игра/игрови дизайн		
Форми на обучение	Преподаване пред класа/групата		
	Самостоятелна работа		
Резюме на	(Мотивация-въведение, прилагане, осмисляне и оценка)		
обучението	Мотивация-Въведение		
	Мотивираме учениците, като играем		
	играта (те не виждат кода). Целта на		
	урока е да се създаде подобна игра.		
	[Стъпка 1]		
	Първата стъпка е да се определи предисторията на играта. Фонът		
	трябва да съдържа различни животни. Имаме три възможности:		
	1. Учениците сами рисуват фона;		
	2. Учениците търсят безплатно изображение онлайн;		
	3. Осигуряваме предварителни ресурси за учениците (ако		
	искаме да спестим време).		
	Учениците вече знаят как да добавят		
	фон на сцена, така че го правят		
	индивидуално.		
	[Стъпка 2]		
	Втората стъпка е да добавите жената фермер. Имаме същите		
	опции като в първата стъпка:		
	1. Учениците сами рисуват жената фермер;		
	2. Учениците търсят безплатно изображение на жената фермер		
	онлайн;		
	3. Предоставяме образ на жената фермер за ученици (ако		
	искаме да спестим време).		





Учениците вече знаят как да добавят нов спрайт, така че го правят индивидуално. [Стъпка 3] След това учениците when 🍋 clicked say Ilyou want to hear the dog, click on the key "D"I for (3) secs трябва да програмират инструкциите за say If you want to hear the cow, click on the key "C"| for (3) secs плейъра. Инструкциите say If you want to hear the sheep, click on the key "S" for (3) secs са дадени от жената

alt 1 secs say If you want to hear the pig, click on the key "P" for (3) secs Учениците valt 1 se say If you want to hear the horse, click on the key "Hill for (3) secs като

използват блокове Say от Looks и Изчакай ... сек.. Учениците вече знаят как да правят това, така че го правят индивидуално.

Изпълнение

фермер.

това,

правят

След това показваме на учениците как да добавят звук в играта. Имаме три възможности:

1. Импортиране на звук от медийна библиотеката на Snap!;

2. Импортиране на звук от нашия компютър чрез плъзгане в Snap!;

3. Запис на нов звук в Snap!

Учителят показва на учениците и трите варианта под формата на фронтално обучение. Когато ги представи всички, учениците започват да програмират поотделно следните задачи (с подкрепата на учителя).

Open Save Save As	^0 ^5	
Save As	minute	
Import		Sounda
Export project.	. (in a new window) as plain text	
pund I Export summar	y	
Export solution	s as pic	
Libraries Backgrounds	Soloct a sou	ind from the media library
of sound		
bund 🖪 rate (4410) 🗢 Hz		
	Export project a Export summar Export summar Export all script Libraries Backgrounds Sounds of sound Dund Erste (44100 - Hz	Export project as plain text Export summary Export summary Export all scripts as pic Libraries Backgrounds of sound out a rate (44100) Hz





Co-funded by the Erasmus+ Programme of the European Union



Учениците трябва да програмират звука на кучето, който ще се възпроизведе при натискане на бутон "D". Те правят това, като от група **Control**, използват блок **Когато клавишът** [the_key] е натиснат и блок Sound / play sound [name_of_sound].



[Стъпка 5]

Учениците трябва да програмират звуци на животни. Първо, те трябва да добавят звуци от компютъра си. Те правят това, като плъзгат звуците в раздела за звуци на фона.

След като импортираме звуците, можем да щракнем с десния бутон върху звуците, за да ги преименуваме. В нашия случай те се наричат: крава, прасе, кон и овца.



След това учениците трябва да добавят звука в скриптове на сцената. Те правят това, като използват *Control / When key is pressed [the_key] и блокиране на звук / възпроизвеждане на звук [name_of_sound]*.







Co-funded by the Erasmus+ Programme of the European Union

Следващата стъпка е да програмирате поздрава на жената фермер. Когато играчът започне играта, жената фермер трябва да каже: "Добре дошли в моята ферма!". Първо,

- 0 -	Farm
€ +	Woman farmer
import a by drago	sound from your computer ng it into here
P	Sound Recorder
	Sound Recorder

учениците трябва да запишат поздрава на жената фермер. Правят го със звукозапис (червен бутон), намиращ се в раздела (Звуци на жената фермер). Когато записват звука, те трябва да го запазят

(бутон Запазване - Save).

След като запазим звука, можем да щракнем с десния бутон върху него, за да го преименуваме. В нашия случай се нарича ферма. Сега учениците трябва да добавят звука в



сценариите на жена фермер. Те правят това, като използват блок

Play sound [name_of_sound]. Play sound tame от група Sound.

[Допълнителна задача]

Ученикът може да надгради фермата, както му харесва, като добави нови спрайтове (фермер, кокошка, трактор, ...) и звуци.

Рефлексия и оценка

Студентите обобщават:

- как са добавили звуци в кода си;
- кои блокове са използвали за вмъкване на звук в кода;
- кои контролни блокове са използвали в своя код;
- защо и как са използвали звукови блокове и контролни блокове.

[Окончателен код]

The woman farmer





	when i clicked ipy sound iams wit i soci is y lyou want to hear the dog.click on the key iDi for i soci is y lyou want to hear the cow,click on the key iDi for i soci is y lyou want to hear the sheep, click on the key iDi for i soci is y lyou want to hear the pig, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the horse, click on the key iDi for i soci is y lyou want to hear the hear the horse, click on the key iDi for i soci is y lyou want to hear the hear
	when he key pressed play sound horse
Инструменти и	• Цяла дейност в Snap !:
ресурси за учителя	https://snap.berkeley.edu/project?user=tadeja&project=Farm
	 Уебсайт с безплатни изображения: <u>https://pixabay.com/</u>
	 Уебсайт на безплатни звуци: https://www.zapsplat.com/
	• Лайович С (2011) Драскотина Научете се програмиране и
	• Vorderman, C. (2017). Racunalnisko nporpamupahe sa dega.
	Люоляна: Мк.
Инструменти и	• Шаблон в Snap!:
ресурси за	https://snap.berkeley.edu/project?user=tadeja&project=Sounds%20o
учениците	f%20the%20farm_0
	• Уебсайт с безплатни изображения: https://pixabay.com/
	• Уебсайт на безплатни звуци: https://www.zapsplat.com/
	• Инструкции за ученика (C4G5_InstructionsForStudent_BG.docx)





Учебен сценарий 6 – Лятната ваканция на хамелеона

Учебен сценарий	Лятната ваканция на хамелеона	
Име		
Предишен опит в	Не се изискват предварителни познания	
програмирането		
Очаквани резултати	Общи резултати о т обучението:	
	 движение на обект въз основа на събития; 	
	• чувствителност към един или повече цветове;	
	 отчитане на булева стойност в логически изрази; 	
	• дефиниране, разграничаване, динамична проверка и	
	реагиране на различни състояния на играта,	
	Специфични резултати от обучението, ориентирани към	
	алгоритмично мислене:	
	• ученикът реализира движението на обекта с клавишите със	
	стрелки, използвайки събития и взема предвид	
	ограниченията,	
	• ученикът използва сензорен цветен блок, за да получи	
	булева стойност за четене на еднократно или многократно	
	сензориране,	
	• ученикът осъзнава, че състоянието на обекта може да бъде	
	изразено с цветовете, които обектът докосва,	
	• ученикът прави разлика между две (основни), пет (пълни)	
	различни състояния и знае как да ги изрази с логически	
	изрази,	
	• ученикът осъзнава, че позицията на обекта се променя	
	динамично и използва цикъл forever, за да проверява	
	многократно текущото състояние,	
	• ученикът използва if , за да даде различни отговори въз	
	основа на текущото положение на обекта.	
Цел, задачи и	Кратко описание: Програмирайте проста игра, в която обектът ще	





кратко описание на	промени костюма си въз основа на цвета на фона.
дейностите	Цел: Учениците ще бъдат запознати със сензорния цветен блок и
	как да го използват в логически изрази, за да разграничат
	динамично променящите се състояния на играта и да дадат
	правилните отговори.
Продължителност	45 минути
Стратегия и методи	Активно обучение, съвместно обучение, решаване на проблеми
на обучение	
Форми на обучение	Фронтално обучение
	Индивидуална работа / работа по двойки / групова работа
Резюме на	(Мотивация-Въведение, Прилагане, Размисъл и оценка)
обучението	Хамелеон замина на лятна ваканция. Той обича да се къпе в
	морето, да релаксира плажа и когато е прекалено горещо, той
	обича да отива в сянката на близките дървета, за да се охлади.
	Тъй като той е хамелеон, той променя цвета си според настоящата
	му позиция.
	[Базова версия]
	В основната версия трябва да правим разлика между две
	състояния.
	[Стъпка 1]
	Молим учениците да редактират фона на сцената, така че той да е
	разделен на две части, син и пясъчен, като всяка представлява
	различно място. Синият цвят е за морето и пясъчен за плажа.
	Можем да инструктираме учениците да включват други
	предмети, за да направят фона по-реалистичен, като: вълни,
	черупки, пясъчни замъци, чадъри и т.н Те трябва да внимават
	да не избират предмети, които са по-големи и изцяло оцветени с
	различни цветове от заден план. В този случай блокът за цветно
	разпознаване няма да може да разпознае в коя част от сцената е
	героят.




[Стъпка 2]

Учениците трябва да нарисуват хамелеон и да оцветят кожата му в два различни цвята:



[Стъпка 3]

Първо трябва да накарат хамелеона си да се движи в четири посоки с помощта на клавиши. Те могат да изберат своя собствена комбинация от клавиши (напр. Клавиши със стрелки или WASD). На този етап предполагаме, че знаят как да го направят от предишни дейности. Трябва да напомним на учениците, че персонажът може да се измести от сцената, ако не използваме подходящ блок при програмиране на движение с блок *bounce if on edge* (отскачане, ако е на ръба).

За да направим движението на хамелеона малко по-реалистично, искаме той да се обърне наляво или надясно към хоризонталната посока, с която сме изправени с блок *point in direction* (посока на движение).







Запознаваме учениците с концепцията за характер, усещащ цвета (цветовете), които той докосва. С блока "touching color?" ("докосващ цвят?") можем да получим информация под формата на булеви стойности - True или False, ако той докосва определен цвят. Тъй като получаваме булева стойност от този блок, можем да я използваме в блока if, където е решено дали ще изпълняваме команди, изброени в тялото му или не. След това обсъждаме със студентите какви са различните позиции на хамелеона на сцената и как можем да ги изразим, използвайки блока touching color?. Има две: Той докосва синия цвят -> Touching color [blue]? Той докосва пясъчния цвят -> Touching color [sandy]? Когато той докосва определен цвят, ние трябва да променим външния му вид и също така да го накараме да каже къде се намира. Можем да променим външния вид на Спрайт, като превключваме между неговите костюми. Това се прави с блока Looks/switch to costume[option], където избираме кой от възможните костюми искаме да покажем. За да накараме хамелеона да говори, използваме Looks/say[text] block. Защото има само две възможности, които можем да използваме блока за условие "if - else". Можем да изберем кой цвят ще проверяваме и по подразбиране другият цвят ще попадне в случай "друго". В примерния код избрахме пясъчен цвят:





	ouching 📙 ?	
say	Pam sunbathing o	n'a beach
swi	tch to costume	kameleon_oranzen -
lse		
say	Pamswimming	-
swi	tch to costume	kameleon_moder

[Стъпка 5]

За ситуации, когато трябва да изпълняваме определени команди за цялото време на програмата, която използваме - цикъл завинаги. Всичко, написано под тялото на цикъл forever, ще се изпълнява отново и отново. Обсъждаме с учениците, че в нашия случай точно това е, което искаме / имаме нужда, за да създадем тази игра.

[Окончателен код]



[Пълна версия]

[Стъпка 1]

Молим учениците да редактират фона на сцената, така че той да е разделен на три части от един и същи цвят, всяка от които представлява различно място: син цвят за морето, пясъчен цвят за плажа и зелен за гората. Те могат да добавят други предмети, за да направят фона по-реалистичен като: вълни, черупки, пясъчни замъци, слънчеви чадъри, дървета и др ... но трябва да внимават добавените предмети да не са по-големи от самия главен герой,





защото в в този случай персонажът няма да докосне нито един от трите цвята и функцията за засичане на Snap няма да може да разпознае в коя част от сцената е героят.



[Стъпка 2]

Те трябва да нарисуват хамелеон и да нарисуват кожата му в пет различни комбинации, представящи позицията му на сцената:



[Стъпка 3]

Първо трябва да накарат хамелеона си да се движи в четири посоки с помощта на клавиши. Те могат да изберат своя собствена комбинация от клавиши (напр. Клавиши със стрелки или WASD). На този етап предполагаме, че те знаят как да го направят от някаква друга дейност. Трябва да предупредим учениците да не забравят, че персонажът може да се измести от сцената, ако не използваме подходящ блок при програмиране на движение (bounce if on edge block).

За да направим движението на хамелеона малко по-реалистично, искаме той да се обърне наляво или надясно към хоризонталната посока, пред която сме изправени (използвайте a *point in direction* block).

when left accur. Any pressed point in direction CO change x by CO if on edge, bounce	when opt around key precord point in direction (23) change z by (23) if on edge, bounce	when down arms a key personal change y hy (15) if on edge, bounce	when is non in y person change y by (53) if on edge, bounce
[Стъпка 4]			





Запознаваме учениците с концепцията за характер, усещащ цвета
(цветовете), който той докосва. С блока "touching color?" можем
да получим информация под формата на булеви стойности - True
or False ако той докосва един или дори няколко цвята в даден
момент. Тъй като получаваме булева стойност от този блок,
можем да я използваме в главата на изречението If, където е
решено дали ще изпълняваме команди, изброени в тялото му или
не.
След това обсъждаме с учениците какви са различните позиции
на хамелеона на сцената и как можем да ги изразим с докосващ
цвят? блок.
Бързо откриваме, че има пет:
1. Той е изцяло от синята част -> Touching color [blue]?
2. Той е между синята и пясъчната част -> Touching
color[blue]? AND Touching color [sand]?
3. Той е изцяло от пясъчната част -> Touching color [sand]?
4. Той е между пясъчната и зелената част -> Touching
color[sand]? AND Touching color [green]?
5. Той е изцяло от зелената част -> Touching color [green]?
Когато той докосва определен цвят (цветове), ние трябва да
променим външния му вид и също така да го накараме да каже
къде се намира. Можем да променим външния вид на спрайта,
като превключваме между неговите костюми. Това се прави с
блока Looks/switch to costume[option], където избираме кой от
възможните костюми искаме да покажем. За да накараме
хамелеона да говори, използваме блока <i>Looks/say[text]</i> .
Първо се грижим за по-простите ситуации, при които хамелеонът
е изцяло в една и съща цветна част на сцената:
touching ? say [any sample blacking ?] switch to costume kameleon modern!





След това формираме логически израз с използването на логически оператор И, защото искаме да проверим дали хамелеонът докосва два цвята едновременно:



Ако комбинираме условните изречения по-горе и ги сложим под блока за събитие *When Green Flag clicked*, забелязваме, че тези условия ще бъдат проверени точно веднъж. Помагаме им да забележат, че тъй като контролираме движението на главния герой, позицията на хамелеон ще се променя през цялото време по време на играта. Ето защо трябва постоянно да проверяваме тези условия не само веднъж, но буквално през цялото време!

[Стъпка 5]

За ситуации, когато трябва да изпълним определени команди за цялото изпълнение на програмата, която използваме - цикъл forever. Всичко, написано под тялото на цикъла forever, ще се изпълнява отново и отново. Обсъждаме с учениците, че в нашия случай точно това е, което искаме / имаме нужда, за да създадем тази игра.

[Окончателен код]







[Учениците коригират кода]

За да опростим тази дейност, можем предварително да подготвим част от кода в шаблонния файл и да инструктираме учениците да го попълнят.

Учениците, които следваха предложената учебна пътека, вече научиха за преместването на обекта с ключове. Така че можем да включим кода за движение в шаблонния файл. Те могат да променят настройките на клавишите от клавишите със стрелки към персонализирано подреждане (например WASD).



За да им помогнем да разберат понятието цикъл forever и как да го използват за откриване на цвят на фона, можем да включим код за откриване на две ситуации: 1) обектът е изцяло на един цвят, 2) обектът докосва два цвята едновременно. Инструктирахме ги да попълнят кода за всеки случай.





	Предложен шаблон за код:	
	when clicked forever if touching ? say l'amisunbathing'on'a beach switch to costume kameleon_oranzen if touching ? and touching ? say l'amibetween:the sea' and the 'beach switch to costume kameleon_oranzen_moder *	
Инструменти и	 Цялата дейност в Snap!: 	
ресурси за учителя	Basic:	
	https://snap.berkeley.edu/project?user=zapusek&project=cha	
	<u>meleon simple</u>	
	Full:	
	https://snap.berkeley.edu/project?user=zapusek&project=cha	
	<u>meleon</u>	
	 Lajovic, S. (2011). Scratch. Nauči se programirati in postani računalniški maček. Ljubljana: Pasadena. 	
	 Vorderman, C. (2017). Računalniško programiranje za otroke. Ljubljana: MK. 	
Ресурси /	• Шаблон в Snap!:	
материали за	https://snap.berkeley.edu/project?user=zapusek&project=cha meleon_template	
студентите	 Полуготов файд в Spapl: 	
	https://spap.berkeley.edu/project?user=zapusek&project=cha	
	meleon half baked	
	 Инструкции за студент (C4G6_InstructionsForStudent_BG.docx) 	





Учебен сценарий 7 – Помогнете на принца и принцесата да открият техните животни

Учебен сценарий	Помогнете на принца и принцесата да открият техните животни	
Име		
Предишен опит в	Добавяне на текст за спрайт	
програмирането	Движение на обект с клавиши със стрелки с помощта на събития	
	Използването на условие за спрайт докосва друг обект.	
	Използване на събития	
Очаквани резултати	Основни очаквани резултати:	
	 Условия за спрайт да докосва даден цвят 	
	• Придвижване до определени координати	
	• Молив горе, Молив долу	
	• Цвят на молива	
	Специфични очаквани резултати свързани с алгоритмично	
	мислене:	
	• Учениците използват оператор if за проверка дали даден	
	обект докосва даден цвят	
	• Учениците задават начални координати на спрайт	
	• Учениците използват блокове "молив горе" и "молив долу"	
	за изчертаване на линия/пътека на движение.	
	• Учениците променят цвета в ззависимост от двойката, която	
	свързва.	
	• Учениците реализират начално изтриване на всички	
	очертавания на сцената от предишно изпълнение на	
	програмата.	
Цел, задачи и	Кратко описание: Момичето трябва да помогне на принцесата да	
кратко описание на	намери котката си, а принцът да намери кучето си. Тя прави това,	
дейностите	като отива при принцесата и й показва, с чертане на линия, пътя до	
	котката си; подобно Момичето показва на принца пътя към кучето	
	му. По този начин Момичето трябва да избягва срещата между	
	животни, така че пътищата им да не се пресичат.	





	Задачи: В първата стъпка учениците трябва да изберат подходящия		
	фон (лабиринт). Те добавят пет спрайта в лабиринта - техния спрайт		
	(момиче), принцеса, принц, котка и куче. След това те програмират		
	движението с ключове (с използване на събития) за момичето,		
	където трябва да обърнат внимание, че спрайтът не ходи по		
	тревата. По-късно те програмират рисуване с писалка и промяна на		
	цвета на писалката със събития. Те също трябва да програмират		
	стартовото събитие, което изчиства пътя и момичето дава		
	инструкции.		
	Цел: Учениците ще бъдат въведени в очертаване на движение на		
	героя с клавиши. Освен това те ще се научат как да използват		
	условни блокове и условия, за да предотвратят движеието по		
	целия екран.		
Продължителност	30 минути		
Стратегия и методи	Активно обучение, обучение, основано на игрови дизайн,		
на обучение	решаване на проблеми		
Форми на обучение	Фронтално обучение		
	Индивидуална работа		
Резюме на	(Мотивация-Въведение, Прилагане, Рефлексия и оценка)		
обучението	Първоначално се дава на студентите:		
	• Фон		
	 Спрайт на момиче 		
	 Код на движение за една посока 		
	Момичето решава да помогне на принцесата да намери котката си,		
	а принцът да намери кучето си, като показва (рисува) пътя към		
	техните животни. За да се избегне объркване, пътеките трябва да са		
	с различни цветове и да не се пресичат.		







[Стъпка 1]

Молим учениците да редактират фона на сцената - лабиринт. За прилагане на "if touching color" ("ако докосваш цвят") или фонът (трева) трябва да е монохромен, или пътеката трябва да има монохромна рамка, както в нашия случай. За да избегнем тези "проблеми" с намирането на подходящ фон, ние им даваме този фон.

[Стъпка 2]

Учениците вече имат момичешкия спрайт в началото. Трябва да намерят още четири спрайта и да ги сложат в лабиринта. За всички спрайтове те трябва да зададат подходящия размер (който е помалък от ширината на пътеките в лабиринта. За всеки спрайт те използват кода:



Препоръчителният размер за момичето е 8%, другите спрайтове могат да бъдат по-големи.

[Стъпка 3]

След това те трябва да направят движението на момичето в четири посоки с помощта на клавиши. Предполагаме, че те вече знаят как





да направят това от предишни дейности. Както и да е, ние им
даваме кода за едно направление, което им помага да направят
още три.
when left arrow - key pressed change y by 10 when left arrow - key pressed change x by -10 when down arrow - key pressed change y by -10
[Стъпка 4]

В следващата стъпка те трябва да предотвратят движението на момичето през поляната. Те правят това, като добавят условен блок, ако докосват кафяв цвят. Ако момичето докосне кафявия цвят (края на пътеката), тя се движи за 10 крачки назад. Не виждаме тези две стъпки и все едно момичето остава на същото място. Това е код за придвижване надясно, така че 10 стъпки назад означава промяна на x с -10.



Те добавят този код под предишния код, напр. за стрелка надясно:



Подобно трябва да се направи и за другите три направления. [Стъпка 5]

След това програмират рисуване. Те правят това чрез блокове pen up и pen down, използвайки събития when key pressed.



Когато се натисне бутон "D" и момичето се премести, тя чертае





линия. Когато се натисне бутон "Е", чертежът спира. По същия начин те задават цвят на писалката, като натискат клавиша.



[Стъпка 6]

Накрая те програмират при щракване на събитие със зеленото знаме, където учениците добавят някои инструкции, които момичето казва в началото.

Когато играете играта, спрете я и я играйте отново, учениците ще видят, че е добре да добавите следните блокове: pen up (в случай, че е останала надолу от предишната игра), clear (изчиства пътя от предишната игра) и преместване до х, у (момичето винаги започва от тези координати, които са вътре в пътеката, а не на тревата).

За да определим началните координати за момичето, хващаме момичето с мишката и го пускаме там, където искаме да започне. След това щракваме върху Motion блок, където можем да намерим х позиция и у позиция. Чрез щракване върху х позиция откриваме х позицията на момичето, подобно на у.







<pre>when i clicked ist size to i fo pen up clear is y izeptine kinese and the kinese to define the intermal for i sec is y izeptine kinese and the kinese to define the intermal for i sec is y izeptine kinese and the kinese to define the intermal for i sec is y izeptine kinese and the kinese to define it when up and with y pressed is y pressed is to define it key pressed is to define it key</pre>
[Допълнителни задачи]
Учениците могат да добавят допълнителни задачи според техните
желания или могат да следват задачите по-долу:
• Задайте начални координати за принца и принцесата и
напишете код за тяхното движение. Задайте подходящия
размер за тях. Те трябва да нарисуват път към своите животни.
 Добавете друг спрайт (животно) за момичето.
 Всеки спрайт трябва да рисува с различен цвят.
• Регулирайте първоначалните инструкции.
• Добавете инструкции за преместване на спрайт и рисуване,
като щракнете върху спрайт. Напр. принцесата казва:
"премествате ме с натискане на клавишите W, S, A И D.
пачертавам потеката с патискале на клавиша 5. Спирам да
котката си!"





Инструменти и	• Цялата дейност в Snap!:	
ресурси за учителя	https://snap.berkeley.edu/project?user=mateja&project=Helpin	
	g%20Prince%20and%20Princess%20to%20find%20their%20ani	
	mals	
	 Дейност в Snap! С допълнителните задачи (възможно 	
	решение):	
	https://snap.berkeley.edu/project?user=mateja&project=Helping%20Pr	
	ince%20and%20Princess%20to%20find%20their%20animals%20	
	<u>%2B%20Add.%20Task</u>	
	• Lajovic, S. (2011). Scratch. Nauči se programirati in postani	
	računalniški maček. Ljubljana: Pasadena.	
	• Vorderman, C. (2017). <i>Računalniško programiranje za otroke</i> .	
	Ljubljana: MK.	
Ресурси /	 Полуготов файл за работа в Snap!: 	
материали за	https://snap.berkeley.edu/project?user=mateja&project=Helping%20Pr	
учениците	ince%20and%20Princess%20to%20find%20their%20animals%20	
	<u>-%20Part</u>	
	• Указания за учениците	
	(C4G7_InstructionsForStudent_BG.docx)	





Учебен сценарий 8 – Рисуване с креда (тебишир)

Учебен сценарий	ий Рисуване с креда (тебишир)	
Име		
Предишен опит в	 Добавяне на текст към спрайт 	
програмирането	 Рисуване с молив (pen up, pen down, set color) 	
	 Преместване на определен брой стъпки 	
	• Използване на цикъл	
	 Използване на събития 	
Очаквани	Основни очаквани резултати:	
резултати	• Цикъл repeat	
	• Завъртане на 90 градуса	
	 Определяне посока на движение (Point in direction) 	
	• Смяна фон на сцена	
	Специфични очаквани резултати, свързани с алгоритмично мислене:	
	• Ученикът използва цикъл repeat когато се повтарят едни и	
	същи блокове няколко пъти	
	• Ученикът използва завъртане на 90 градуса когато изчертава	
	различни фигури (квадрат, правоъгълник, буква "Т")	
	• Ученикът описва заначението на установяване на посока на	
	движение 90	
	• Ученикът сменя фон на сцена при използване на събитието	
	when a key is pressed	
Цел, задачи и	Кратко описание: Играчът получава три различни фона и трябва да	
кратко описание	свърже точки в три различни форми - квадрат, правоъгълник и буква	
на дейностите	"T".	
	Задачи: Учениците избират фона "дъски" и започват с рисуване на	
	квадрат. Тяхната изходна позиция е точката "А". Когато рисуват	
	квадрат, те повтарят определени стъпки 4 пъти, така че вместо да	
	пишат същия код 4 пъти, те могат да използват повторение на	
	цикъла 4 пъти. След това те нарисуват правоъгълник, също с	





	използване на повторение на цикъл, този път повторете 2 пъти. В		
	последната си задача те трябва да свържат точки във формата на		
	буквата "Т", където трябва да открият броя на стъпките. Те могат да		
	използват повторение на цикъла, когато е възможно.		
	Цел: Учениците ще бъдат запознати с рисуването на различни		
	фигури с код. Те ще се научат да използват повторение на цикъл,		
	за да съкратят кода и да променят фона.		
Продължителност	60 минути		
Стратегия и	Активно обучение, обучение, основано на игрови дизайн, решаване		
методи на	на проблеми		
обучение			
Форми на	Фронтална работа		
обучение	Индивидуална работа / Работа по двойки		
Резюме на	(Мотивация-Въведение, Прилагане, Рефлексия и оценка)		
обучението	Първоначално се дава на учениците:		
	• Три фона с всички точки, които трябва да свържат		
	• Крейда спрайт		
	Кредата трябва да нарисува квадрат, правоъгълник и да свърже		
	точки във формата на буква "Т", но не знае как да се движи и как да		
	се завърти. Напишете код и покажете на тебешира как се прави!		
	[Стъпка 1]		
	URAW A SOUARE		
	, D		
	A		
	учениците започват с този фон. те пишат код за рисуване на квадрат.		
	Започвайки от точка "А", те преместват Х стъпки към точка "В",		





завъртат 90 градуса вляво, преместват Х стъпки до точка "С", завъртат 90 градуса вляво, преместват X стъпки до точка " D ", завъртете 90 градуса вляво, преместете Х стъпки до точката, А "(и завъртете 90 градуса вляво). move (150) steps alt 🕦 secs turn (90 degrees 1 3 move (150) steps turn (90) degrees alt 1 s move (150) steps turn () (90) degrees valt (1) secs move 150 steps ait 1 secs turn 5 90 degrees wait (1) secs Използвайки turn 90 degrees е най-лесният начин, тъй като винаги можем да използваме завъртане на 90 градуса (зависи само дали искаме да завием наляво или надясно). Използвайки Using point in direction 0, 90, 180, -90 е друга опция, но е малко по-сложна, защото трябва да отделим 4 възможности и не можем да използваме цикъл repeat. Блокът Wait 1 secs блок се добавя само за да видите чертежа / всички стъпки. Без този блок целият код се случва за секунда. Учениците трябва да го пробват без този блок, за да разберат значението му. Питаме ученика как би съкратил кода, ако е възможно. Има ли част от повторенията? Отговорът е да. Вместо да пишем същия код 4 пъти, при програмирането използваме цикъл repeat.





move (150) steps wait (1) secs turn (50) degrees wait (1) secs

Ако искаме да видим какво нарисуваме, трябва да поставим блок *pen down* преди цикъла *repeat*.



Ако искаме тебеширът да не се върти при завъртане, щракваме върху блока don't rotate in direction.



[Стъпка 2]

За активиране на кода учениците използват блока за събития, напр. when S key is pressed. Те могат да изпозлват също set pen color, и както вече знаят от предходната тема да изпозлват следните блокове pen up (в случай, че е останал настрана от предишното проиграване), clear (изчиства рисунката от предишното възпроизвеждане) и go to x, y (тебеширът винаги започва от тези координати).

Понякога се случва да спрем програмата по време на пиесата и след това спрайтът да се завърти в "странна посока". Това е проблем при стартиране на играта отново, ако спрайтът се завърти погрешно, той ще отиде например надолу, а не вдясно на първата стъпка. За да избегнем този проблем, добавяме блок *point in direction 90*.













За да напишат код, учениците трябва да щракнат върху *Scripts*. За да програмират променящия се фон, те избират блок от Събития - *when R key pressed* и след това *switch to costume boardRectangle*.



b) Щракваме обратно върху тебешира.













when 1 key pressed

b) Те щракват обратно върху тебешира и под кода от [Стъпка 4] добавят блок, където ще казват на играч какво да направи, за да смени фона, което е, натиснете клавиша "Т".



[Стъпка 6]



След натискане на клавиша "Т", фонът се променя на този. Подобно на преди, те трябва да свържат точки и да нарисуват буква "Т". Учениците могат да копират предишните блокове от кода и да ги коригират.

Учениците ще трябва да сменят началните координати, които не са същите като преди. Те вече знаят как да определят правилните координати от предишната дейност.

След това те пишат код за изчертаване на буква "Т". Те трябва да открият броя на стъпките. Едно от възможните решения е:







a) Те щракват върху фона (наименована board, от дясната страна на екрана), където пишат код за промяна на фона. Те ще направят това с when S key pressed и след това switch to costume boardSquare.



b) Те щракват обратно върху тебешира и под кода от [Стъпка 6] добавят блок, където ще казват на играч какво да направи, за

[Стъпка 7]





да смени фона, което е, натиснете клавиша "S".		
say Press Sitorstart from the beginning. for (2) secs		
[Финален код]		
[Φιμαπεμ κομ] when s key pressed set pen color to clear pen up point in direction 90 g go to x: 00 y: 005 pen down repeat 4 move 150 steps wait 1 secs say Press:Rito-continue; for 2 secs when r key pressed set pen color to clear pen up point in direction 90 g go to x: 00 y: 005 pen down repeat 2 move 15 steps wait 1 secs turn 90 degrees wait 1 secs turn 90 degrees	<pre>when [= key pressed set pen color to clear pen up point in direction 90 go to x: -55 y: -138 pen down move 60 steps wait 1 secs turn 90 degrees wait 1 secs turn 90 steps wait 1 secs turn 90 degrees wait 1 secs turn 90 degrees wait 1 secs turn 90 degrees turn 90 degree</pre>	
[Допълнителни задачи]		





	Учениците могат да добавят допълнителни задачи според техните
	желания или могат да следват задачите по-долу:
	 Добавете нов фон и нарисувайте няколко точки.
	• Напишете код, който свързва точките. Можете да нарисувате
	фон или да използвате даден.
Инструменти и	 Цялата дейност в Snap!:
ресурси за учителя	https://snap.berkeley.edu/project?user=mateja&project=Drawin
	g%20with%20a%20chalk
	• Lajovic, S. (2011). Scratch. Nauči se programirati in postani
	računalniški maček. Ljubljana: Pasadena.
	• Vorderman, C. (2017). <i>Računalniško programiranje za otroke</i> .
	Ljubljana: MK.
Инструменти и	 Полуготови ресурсни материали в in Snap!:
ресурси за	https://snap.berkeley.edu/project?user=mateja&project=Drawin
учениците	g%20with%20a%20chalk%20-%20Part
	• Инструкции за учениците
	(C4G8_InstructionsForStudent_BG.docx)





Учебен сценарий 9 - Прибиране на боклука и почистване на парка

Учебен сценарий	Прибиране на боклука и почистване на парка
Име	
Предишен опит в програмирането	 Задаване на начални координати Задаване размер на спрайта Добавяне на текст към спрайта Придвижване на обект със стрелки, използвайки събития Използване на условия за проверка дали даден обект докосва друг обект
Очаквани	Общи очаквани резултати:
резултати	 Променливи Показване и скриване на спрайтове Дублирани спрайтове Дублиран блок от код Условия
	Специфични очаквани резултати за развитие на алгоритмично мислене:
	 ученикът използва променлива за преорояване на събраните отпадъци Ученикът използва скрит спрайт, когато е докоснат спрайт и показва спрайт в началото
	 Ученикът знае как да дублира спрайт (от една бутилка до например 4 бутилки) Ученикът знае как да дублира блок код (от спрайт на бутилка на хартион спрайт)
	 Ученикът знае как да използва условни условия за проверка дали е показан спрайт и дали всички боклуци са взети
Цел, задачи и	Кратко описание: Паркът е пълен с боклук и момичето решава
кратко описание	да го почисти. Когато събере всички боклуци, тя ги хвърля в
на дейностите	кошчето.
	Задачи: Учениците започват с задаване на начални координати
	за момичето. Играта приключва, когато момичето събере всички
	боклуци и ги сложи в кошчето. За целта учениците ще трябва да
	използват променливи за преброяване на точки (1 събрано
	кошче = 1 точка). Когато момичето докосне кошчето, го вдига,
	кошчето се скрива и броят точки се увеличава за 1. Когато вземе
	цялото кошче, тя отива до кошчето. Ако тя не вземе целия боклук





	и отиде по-рано в кошчето, кошчето казва да се върне, когато
	вземе цялото кошче.
	Цел: Учениците ще научат как да използват променливи и как
	да дублират блок код или дори цял спрайт.
Продължителност	45 минути
Стратегия и	Активно обучение, обучение, основано на дизайн на игри,
методи на	решаване на проблеми
обучение	
Форми на	Фронтално обучение
обучение	Индивидуална работа
Резюме на	(Мотивация-Въведение, Прилагане, Рефлексия и оценка)
обучението	Първоначално се дава на студентите:
	 Фон Спрайт за момичета (с кода за движение), спрайт за бутилки, спрайт за хартия и спрайт за боклук Момичето иска да се разходи и да се наслади на деня си в парка.
	Когато идва там, тя вижда, че паркът е пълен с боклук. Тя решава
	да вземе всички боклуци. Когато го направи, най-накрая може да
	легне и да се наслади на слънчевия ден в чист парк.















Ако квадратчето е отметнато, променливата с нейната стойност ще се вижда на екрана:



В началото на играта стойността на променливата трябва да бъде 0, тъй като няма прибран боклук. Под кода от [Стъпка 1] студентът добавя блок set ___ to 0. Чрез щракване върху падащото меню те избират подходяща променлива, която е points.

set points to D

[Стъпка 3]

Учениците пишат код за бутилка. Идеята е, че спрайтът изчезва (което означава скрий), когато докосне момичето.

Така кодът ще започне, когато спрайтът докосне момичето. Тогава трябва да помислим в кой случай тя взема боклука. Ако казахме, че кошчето се скрива, когато е взето, можем да го вземем само ако все още е там = е показано. Ако спрайтът





(бутилката) все още е там, ние го вдигаме "и го поставяме в полето с променливи". Преди имахме 0 елемента в променливата *points*, сега имаме 1. Виждаме, че като вземем боклука, променяме броя на променливите (*points*) с 1, което е, увеличете с 1. Когато боклукът бъде взет, ние го скриваме.



Сега можем да тестваме дали нашият код е правилен.

Щракаме върху зелено знаме и взимаме бутилката. Бутилката трябва да изчезне и броят точки трябва да е 1. След това искаме да играем отново и отново щракваме върху зеления флаг. Какво става? Къде е бутилката сега?

Бутилката е скрита, ние я скрихме преди. Така че в началото на играта трябва да програмираме, така че бутилката да се показва. Правим това, като изберем блок Show (покажи).



[Стъпка 4]

Сега учениците искат да имат повече бутилки в играта си, за да могат лесно да дублират своя спрайт. Те щракват с десния бутон върху спрайта и избират duplicate (дублирай).



Сега те просто щракват с мишка върху новата бутилка и я плъзгат някъде вътре в лабиринта. Те могат да повторят тази стъпка и да дублират бутилката отново.





[Стъпка 5]

Сега учениците искат да имат същия код за хартиения спрайт. Те могат да дублират кода на бутилката, като щракнат с десния бутон върху блока с код:



И го пуснете в хартиения спрайт, като щракнете с мишката върху спрайта за хартия.



Те повтарят тази стъпка, за да дублират блока when green flag clicked – show.

Те могат също да повторят [Стъпка 4] и да дублират целия хартиен спрайт, за да има повече боклук от хартия в лабиринта. [Стъпка 6]

Последното нещо, което учениците трябва да направят, е да напишат код за кошчето. Кофата за спрайт вече е дадена, те могат да я преместят навсякъде вътре в лабиринта.

Също така този код ще се активира, когато момичето го докосне. Кошчето ще трябва да провери дали всички боклуци са взети. Благодарение на променливите точки това ще бъде лесно да се направи. Да приемем, че имаме 8 спрайта в играта, така че учениците трябва да проверят дали броят на точките е равен на 8. Ако е, това означава, че всички боклуци са взети, в противен случай не е така. Те ще използват инструкция if, за да





програмират това и ще добавят текст, за да кажат на играча дали
е взел всички боклуци или не.
when touching Gif ? If points = 8 say Congratulations! You picked up all the trash! for 2 secs else say Come back when you pick up all the trash! for 2 secs [Финален код]
Момиче
<pre>when clicked ge to x: 189 y: 155 say Pickup all the trash and public in the trash can for 2 secs set points to 0 when up arrow = key pressed point in direction ① move 10 steps if touching 2 move 10 steps if touching 3 move 10 steps if touching 4 mov</pre>
Бутилка/ Хартия
when touching Gid = ? If shown? change points = by 1 hide





	Кошче when touching Girl ? If points = 3 say Congratulations! You picked up all the trash! for 2 secs else say Come-back when you pick up all the trash. for 2 secs
	[Допълнителни задачи] Учениците могат да добавят допълнителни задачи според техните желания или могат да следват задачите по-долу:
	 Добавете друг вид отпадъци (например биоотпадъци). В кошчето пише напр. "Взехте Х бутилки, Y хартии и Z дини". Ако играч вземе цялото кошче, кошчето казва: "Поздравления! Взехте всички боклуци! " Ако даден играч не вземе цялото кошче, кошчето му казва кой боклук не е бил взет, напр. "Не сте взели всички бутилки. Не сте взели всички дини. и "Върнете се, когато вземете всички боклуци".
Инструменти и ресурси за учителя	 Цялата дейност в Snap!: <u>https://snap.berkeley.edu/project?user=mateja&project=Pick</u> ing%20up%20trash%20and%20cleaning%20the%20park Допълнителни задачи (възможни решения): <u>https://snap.berkeley.edu/project?user=mateja&project=Pick</u> ing%20up%20trash%20and%20cleaning%20the%20park%20 %2B%20Add.%20Task Lajovic, S. (2011). Scratch. Nauči se programirati in postani računalniški maček. Ljubljana: Pasadena. Vorderman, C. (2017). Računalniško programiranje za otroke. Ljubljana: MK.
Инструменти и ресурси за учениците	 Полуготови файлове с ресурсни материали в Snap!: <u>https://snap.berkeley.edu/project?user=mateja&project=Pick</u> <u>ing%20up%20trash%20and%20cleaning%20the%20park%20-</u> <u>%20Part</u> Инструкции з аучениците (C4G9_InstructionsForStudent_BG.docx)





Учебен сценарий 10 - Хранене на котките

Учебен сценарий	Хранене на котките
Име	
Предишен опит в	 Условни блокове (if, if-else blocks)
програмирането	 Извеждане на текст (block say)
Очаквани резултати	Основни очаквани резултати:
	 Задаване и учеличаване стойност на променлива, присвояване стойност на променлива вътре / извън цикъла,
	 цикъл for (repeat n times),
	 случайни числа,
	• слепване на стрингове,
	• логически и аритметични оператори,
	• вход
	Специфични цели на обучение ориентирани към алгоритмичното
	мислене:
	 Ученикът разпознава ситуацията за използване на повторение п цикъл пъти, Ученикът прави разлика между присвояване на стойността във всяка итерация на цикъла и веднъж преди цикъла. Ученикът използва блок за въвеждане, за да получи номера от играч, Ученикът знае как да използва аритметични оператори, за да генерира правилния отговор, Ученикът използва if - else, за да провери правилността на въведеното от играча и дава подходящ отговор, Ученикът знае как да използва променлива, за да брои правилни отговори.
Цел, задачи и	Кратко описание: Програмирайте игра, в която играчът ще трябва
кратко описание на	да извърши десет изчисления за умножение и да преброи
дейностите	верните отговори.
	Задача: Програмирайте дейността, при която пазачът на приюта
	Марта многократно ще пита играча за броя на котките, които
	може да храни в определена стая. Броят зависи от броя и размера
	на купите. За всяка стая тези два номера трябва да бъдат





	разпределени произволно. Трябва да имаме и брояч, който да
	отчита правилните отговори. Първо пазителят на приюта трябва
	да обясни заданието за играча и след това играта започва. Играта
	приключва, когато тя поиска броя на котките 10 пъти. Всеки път тя
	трябва да даде отговор дали входният номер е правилен или не.
	След активност тя трябва да обобщи колко успешен е бил играчът,
	тя казва колко пъти играчът е отговорил правилно и колко пъти е
	сгрешил.
	Учениците ще бъдат запознати с концепцията за многократно
	присвояване на случайни стойности в цикъл и как тя е различна
	от тази, когато го правим извън цикъл. Те също така ще научат
	как да получат, тестват и преброят правилните входове на
	играчите.
Продължителност	45 минути
Стратегия и методи	Активно обучение, съвместно обучение, решаване на проблеми
на обучение	
Форми на обучение	Фронтално обучение
	Индивидуална работа / работа по двойки / групова работа
Резюме на	(Мотивация-Въведение, Прилагане, Рефлексия и оценка)
обучението	Пазителят на приюта се опитва да нахрани котките си в десет
	различни стаи. Във всяка стая има произволен брой купи (2 до
	10), които имат различни размери (1 до 5), но във всяка стая
	всички купи са с еднакъв размер. Размерът на купата показва
	колко котки могат да ядат от нея, например ако размерът на
	купата е 3, това означава, че 3 котки могат да ядат от нея.
	Помогнете да намерите броя на котките, които тя може да храни
	във всяка стая.
	[Стъпка 1]
	Първо инструктираме учениците да проектират интересен фон за
	играта. Ако искаме да спестим време, можем да им го
	предоставим.




Co-funded by the Erasmus+ Programme of the European Union



[Стъпка 2]

Трябва да изберем нов костюм за спрайта на костенурките по подразбиране, който ще представлява пазителя на приют за котки.



[Стъпка 3]

За да съхраним необходимите стойности, са ни необходими три променливи: 1) за съхраняване на броя на верните отговори, 2) за присвояване на произволна стойност за броя на купите във всяка стая (2-10) и 3) за присвояване на произволна стойност за купата капацитет (1-5). Началната стойност на точния брояч на отговори ще трябва да бъде 0, а другите два не трябва да бъдат инициализирани преди цикъла, защото ние ще им присвоим нови произволни стойности във всяка итерация на цикъла. Ние също искаме да броим стаите, но не се нуждаем от специална променлива, за да го броим. Ще използваме същата променлива като в цикъл for. Неговият номер ще бъде инициализиран до стойност 1 и след това ще бъде увеличен с

1 за всяка итерация, докато се достигне стойност 10. Това възпроизвежда броенето на стаята.



[Стъпка 4]

След това трябва да програмираме инструкциите за плейъра. Правим това с *Looks / say [string]* и *wait [n] seconds*.





Co-funded by the Erasmus+ Programme of the European Union

say Inimy shelter there are 10 rooms. for (3) secs
say Inteachtroom/Ithaverdifferent/number/of/bowls,/at/least/2/and/never/more/than/10. for (6) secs
say Allthe'bowls'in'one'room'are'the'same'size. for (4) secs
say But'different'rooms'have'different bowl'sizes! for 4 secs
say Bowlisizesicanibeifromi1itoi5, for (4) secs
say If bowlisize is 3,1 can feed 3 cats with it. for (6) secs
say Please help merfind the number of cats locan feed in each room! for (5) secs

[Стъпка 5]

Обсъждаме с учениците какви са действията, които ще се случат във всяка стая и по този начин ще бъдат еднакви. Това са команди, които ще трябва да бъдат поставени вътре в блока на цикъла, за да бъдат изпълнени във всяка итерация на цикъла.

Първо ще трябва да зададем произволно стойност (1-10) за броя на купите и размера на купата в тази стая (1-5). Тогава ще трябва да попитаме играч колко котки можем да храним в тази стая. Нейният отговор ще трябва да бъде тестван за коректност и ние ще трябва да дадем подходящ отговор и да запомним дали е бил верен (брояч на верните отговори). В края на всяка итерация също ще трябва да увеличим номера на стаята с 1.

[Стъпка 6]

За да присвоим произволно стойностите за броя на купите и техния размер, ще използваме стойност Variables / set [options] с Operators/pick random [n] to [m].



[Стъпка 7]

Искаме да поискаме от играча броя на котките, които можем да храним в блока Sensing / ask [string] и изчакайте, защото в противен случай той ще се покаже за определени секунди и след това ще бъде актуализиран с нов ред текст. По този начин играчите могат лесно да забравят колко купи / размери са в





текущата стая. За да направим низ, който ще бъде изграден от комбинация от текст и препратки към променливи, които използваме Operators/join [string1][string2] block. Ще трябва да разширим този блок, така че да отговаря на цялото изречение.

join Theretare: number_of_bowls) thowis. The bowl size is: bowl_size

[Стъпка 8]

Трябва да поставим този дълъг низ вътре Sense/Ask [string] и блока wait block за да получим отговор от играча.

ask join There'are: number_of_bowls !bowls.'The'bowl'size'is: bowl_size an

[Стъпка 9]

Когато играчът отговори, трябва да проверим верността. Има само две възможни ситуации, играчът може да бъде правилен или грешен, така че ще използваме блока If-Else. Правилният отговор е стойността на умножаване на броя на купите с размера на купата. Трябва да проверим дали отговорът на играча е равен на това число. Ако отговорът е верен, увеличаваме брояча на верните отговори с 1 и даваме отговор. Ако не, ние само даваме отговор. Не е нужно да броим грешни отговори, защото можем да го изчислим от правилния брояч на отговорите.



[Стъпка 11]

Сега трябва да изберем цикъл. Както споменахме по-рано, найдобре е да изберете за цикъл, защото променливата, която се използва за итерация, възпроизвежда преброяването на стаите.





[Стъпка 12] Когато цикълът спре, играта приключва. Ние предоставяме информация за постиженията на играчите. Броят на верните отговори се съхранява в брояча на верните отговори; броят на грешните отговори може да бъде изчислен. [Финален код] when 🚬 clicked set correct_answers = to 0 say Introvishelter therefare 10 rooms. for (3) secs say Inteachtroom/lihave/differentinumber/of/bowls,tat/least/2/and/never/more/than/10. for 6 say Alltherbowlstintonetroomtarethetsametsize. for (4) secs say But different rooms have different bowl sizes! for (4) secs say Bowlisizes/can/be/from/1/to/5. for (4) secs say The size of the bowl tells us how many cats we can feed with it. for (6) secs say If bowl/size is 3,1 can/feed 3 cats with it. for (6) secs say Please help metfind the number of cats loan feed in each room! for (5) secs for (i) = (1) to (10) set number_of_bowls to pick random (2) to (10) set bowl_size to pick random 1 to 💿 say join Inthercom: in the for 2 secs Theretare: number_of_b vls 'bowls.'The'bowl'size'is:' bowl_size ask join and . How many cats can i feed? wait if answer = (number_of_bowls) × (bowl_size change correct_answers by 1 say Great!'Your'answer'is'correct! for 2 secs else say This/is/not/the/right number/of/cats. for (2) secs say join The'correct'answeris: number_of_bowls × bowl_size Icats. 📢 for 2 secs if (i) < 10 say Try'to'guess'the'right'number in the next room! for 2 secs say The'game'is'over. for 2 secs time(s). () for (5) secs say join You answered correctly: say join You'were'wrong: (10) - correct_a time(s). () for 5 secs





	[Основна версия на дейността]
	За да спестим време, можем да използваме основната версия на
	сценария. В основна версия са включени всички основни понятия,
	други функции, описани по-горе, могат да бъдат използвани като
	по-късни надстройки.
	when a clicked
	set correct_answers to 0
	repeat 10
	set number_of_bowls to pick random 2 to 10
	set bowl_size = to pick random 1 to 5
	ask join There are: number_or_bowns, there ownsizens, bown_size and
	if answer = number_of_bowls × bowl_size
	change correct_answers by 1
	say Greatl'Your'answeris'correct for 2 secs
	say Thisisinotitheiright number of cats. for (2) secs
	say the game is over, for (2) secs
	say join You'were'wrong: 10 - correct_answers time(s). () for 5 secs
Инструменти и	 Цялата дейност в in Snap!:
ресурси за учителя	<u>https://snap.berkeley.edu/project?user=zapusek&project=cat</u> <u>feeding 2</u>
	• Lajovic, S. (2011). Scratch. Nauči se programirati in postani
	računalniški maček. Ljubljana: Pasadena.
	 Vorderman, C. (2017). Računalniško programiranje za otroke. Ljubljana: MK.
Инструменти и	• Шаблон в Snap!:
ресурси за	https://snap.berkeley.edu/project?user=zapusek&project=cat_feedin
учениците	<u>g_template</u>
	• Инструкции за учениците
	(C4G10_InstructionsForStudent_BG.docx)





Учебен сценарий 11 - Познай броя на котките в приют

Учебен сценарий	Познай броя на котките в приют
Име	
Предишен опит в	 Условни блокове (if block)
програмирането	 Извеждане на текст (block say)
Очаквани резултати	Основни очаквани резултати:
	 Случайни числа, Присвояване стойност на променлива,
	• Вход,
	 Цикъл repeat until,
	• Оператори за сравнение,
	• Брояч
	Специфични очаквани резултати, ориентирани към алгоритмично
	мислене:
	 ученикът присвоява произволна стойност на променливата, ученикът използва блок за въвеждане, за да получи номера от играч, ученикът използва повторение до цикъл, за да поиска многократно играч да въведе числото и да извърши тестване на стойност ученикът извършва тестване на стойност с оператори за изречение и сравнение и дава подходящ отговор, ученикът задава условието на цикъла за повторение, за да тества дали играта е приключила, ученикът осъзнава, че не трябва да тества дали играта е приключила, тъй като тя е намерена в състояние, ученикът прилага брояч за преброяване на предположенията на играчите и използва крайната стойност, за да направи разлика между двата възможни резултата.
Цел, задачи и	Кратко описание: Програма на проста игра, в която в началото
кратко описание на	произволно число от 1 до 100 ще бъде разпределено на случаен
дейностите	принцип към променлива. Играчът ще се опита да го отгатне, като
	въведе цифри. Тя ще получи отговор, ако входното число ще
	бъде: повече, по-малко или равно на случайната стойност.





	Задача: Програмирайте приюта за котки Марта да задава на
	случаен принцип броя на котките, да попита играча за него или
	името му и след това да му обясни задачата. След това Марта
	трябва да поздрави играча с неговото / неговото име и след това
	многократно да поиска номер. Когато играчът въведе своето
	предположение, тя трябва да отговори: 1) ако входният номер е
	по-малък от действителния брой, тя казва: "броят на котките е по-
	голям", 2) ако входният номер е по-голям от действителния брой,
	казва тя : "Броят на котките е по-малък", 3) ако входният номер е
	правилен, тя казва: "Отлично, познахте правилния номер".
	Програмирайте брояч, който ще отчита всеки опит на играч.
	Когато играчът отгатне правилния номер, трябва да проверите
	дали броят на опитите е по-малък от 5. В този случай играчът
	получава котката, в противен случай не.
	Цел: Учениците ще бъдат въведени да повтарят до цикъл и как
	да зададат условието да имплицитно проследява условието,
	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват
	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна
	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите.
Продължителност	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути
Продължителност Стратегия и методи	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми
Продължителност Стратегия и методи на обучение	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми
Продължителност Стратегия и методи на обучение Форми на обучение	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми Фронтално обучение
Продължителност Стратегия и методи на обучение Форми на обучение	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми Фронтално обучение Индивидуална работа / работа по двойки / групова работа
Продължителност Стратегия и методи на обучение Форми на обучение Резюме на	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми Фронтално обучение Индивидуална работа / работа по двойки / групова работа (Мотивация-Въведение, Прилагане, Рефлексия и оценка)
Продължителност Стратегия и методи на обучение Форми на обучение Резюме на обучението	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми Фронтално обучение Индивидуална работа / работа по двойки / групова работа (Мотивация-Въведение, Прилагане, Рефлексия и оценка) Пазачът на приюта за котки Марта иска да познаете точния брой
Продължителност Стратегия и методи на обучение Форми на обучение Резюме на обучението	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми Фронтално обучение Индивидуална работа / работа по двойки / групова работа (Мотивация-Въведение, Прилагане, Рефлексия и оценка) Пазачът на приюта за котки Марта иска да познаете точния брой котки, които тя има в приюта си. Числото може да бъде между 1 и
Продължителност Стратегия и методи на обучение Форми на обучение Резюме на обучението	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми Фронтално обучение Индивидуална работа / работа по двойки / групова работа (Мотивация-Въведение, Прилагане, Рефлексия и оценка) Пазачът на приюта за котки Марта иска да познаете точния брой котки, които тя има в приюта си. Числото може да бъде между 1 и 100. Играчът въвежда числото, а тя отговаря, ако текущото число е
Продължителност Стратегия и методи на обучение Форми на обучение Резюме на обучението	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми Фронтално обучение Индивидуална работа / работа по двойки / групова работа (Мотивация-Въведение, Прилагане, Рефлексия и оценка) Пазачът на приюта за котки Марта иска да познаете точния брой котки, които тя има в приюта си. Числото може да бъде между 1 и 100. Играчът въвежда числото, а тя отговаря, ако текущото число е по-малко, повече или равно на правилния брой котки. Ако играч
Продължителност Стратегия и методи на обучение Форми на обучение Резюме на обучението	да зададат условието да имплицитно проследява условието, което спира играта. Те също така ще се научат как да използват променливи в различни ситуации: да зададат произволна стойност, като брояч или да получат вход на играчите. 45 минути Активно обучение, съвместно обучение, решаване на проблеми Фронтално обучение Индивидуална работа / работа по двойки / групова работа (Мотивация-Въведение, Прилагане, Рефлексия и оценка) Пазачът на приюта за котки Марта иска да познаете точния брой котки, които тя има в приюта си. Числото може да бъде между 1 и 100. Играчът въвежда числото, а тя отговаря, ако текущото число е по-малко, повече или равно на правилния брой котки. Ако играч отгатне броя на котките за по-малко от пет опита, тя получава





[Стъпка 1]

Първата задача е да се направи интересен фон за играта. Студентите могат да го нарисуват сами или да използват безплатни изображения за лиценз от интернет. За да спестим време, можем да подготвим фона предварително.



[Стъпка 2]

Трябва да изберем нов костюм за спрайта на костенурките по подразбиране, който ще представлява пазителя на приют за котки.



Обсъждаме с учениците, че тази игра може да бъде интересна, ако я играете повече от веднъж, ако броят на котките е зададен на случаен принцип. За да запазим това произволно число за сравнение на числата за предположения, ние също трябва да го съхраним в променлива. Сега променливите (предполагаме, че все още не познават концепцията за списъци) са единственият начин да запомните определена стойност в Snap. Това трябва да се случи, когато програмата стартира (Event/When green flag is clicked).

when clicked

[Стъпка 4]

Пазителят на приюта пита играча за името й, за да я поздрави. Това се прави с Sense / ask [низ] и блок за изчакване. Отговорът на





играча се съхранява във вградена променлива с име отговор. За да я поздравим, трябва да се присъединим към низа, съхраняван в променливата отговор, с някакъв поздрав. Това се прави с блок Operators / join [string1] [string2]. За да покажем текста, използваме Изглежда / казва [низ] за п секунди блок. Също така използваме тези блокове, за да напишем инструкции за игра. Също така можем да подчертаем, че е важно да бъдете внимателни към продължителността на показване на текста.



[Стъпка 5]

Обсъждаме с учениците, че не е възможно да се предскаже колко пъти играчите ще трябва да познаят, за да намерят правилния номер. Тя може да извади голям късмет и да го познае при първия си опит, може би ще са й необходими 5 предположения или дори повече, не можем да кажем! Това е причината да трябва да изберем правилния цикъл за дадената задача. Пазителят на приюта трябва многократно да поиска номер и да даде подходящ отговор, докато играчът не познае правилния номер. Единственият цикъл, който можем да реализираме желаното изпълнение, се повтаря до цикъл [condition]. Условието е сравнително лесно да се види, трябва да го циклираме, докато отговорът на играча, който се съхранява във вградения променлив отговор, е равен на стойността, съхранявана в променливата *cat_number*.

[Стъпка 6]

antil answer = number_of





След това трябва да попитаме учениците кои са командите, които ще влязат в тялото на цикъла. Каква е активността или командите, които ще се повтарят, докато играчът не познае правилния номер? Първо, трябва да помолим играча да въведе число, след което трябва да отговорим въз основа на стойността на това число.



[Стъпка 7]

Последното нещо, което трябва да се обясни или обсъди с учениците, е кога ще приключи този цикъл и какво означава това. Когато отговорите на играчите ще бъдат равни на броя на котките, и двете условия в тялото на цикъла ще бъдат неверни, така че цикълът ще премине в следващата итерация, проверявайки състоянието на цикъла. Този път условието ще бъде вярно, така че цикълът ще се прекрати и командите, които следват цикъла, ще бъдат изпълнени. Перифразирайки, когато цикълът се прекрати, знаем, че играчът е отгатнал правилно числото. Така че сега можем да отговорим съответно.







[Стъпка 9]

Трябва да създадем нова променлива, която ще има ролята на брояч, и да я инициализираме до стойността 0. Обсъждаме с учениците значението на инициализирането на променлива и разликата между задаването на стойността и увеличаването й. Когато зададем стойността на променлива, предишната стойност се губи. Това не е добре за брояч. Ако увеличим стойността на променливата с някакво число, добавяме това число към каквато и да е променлива на стойността по-рано. Точно това искаме в тази ситуация. Всеки път, когато играч въведе ново число, ние искаме да го увеличим с 1.

[Стъпка 10]

След верния отговор трябва да проверим стойността на променливата брояч, за да решим дали играчът ще вземе котката или не. Тъй като Snap има само логически оператори по-малко (<) и няма оператори по-малко или равно, условието за решаване дали играчът получава котката е cat_counter <6. Това също е добър пример за използване на блока на условията lf-Else, защото

83





ние разграничават двата случая.
[Краен код]
when clicked set ty_counter to 0 set number of cats to pick random 1 to 100
say Hirliam Martha. Cat Sheltenkeeperl for (2) secs
ask Whatisyourname? and wait
say join 10 name 10 for 20 secs
say Havearlittlergamerforyourtoday for 3 secs
say Ifyour canquess how many cats do Phave in my shelten in 5 tries. for (4) secs
say [willgiveryourgne] for (2) secs
repeat until answer = number of cats
ask Howmany cats do your think I have? and wait
change try_counter by 1
if answer < number_of_cats
say No.no.Phavemore/Cats/than/that/ for (2) sets
il answer > number_of_cats
say interelessicats. for #2, secs
say Awsomell Yourguessed the right number of cats! for (2) secs
il try_counter < 3
say Becauseryourdid it in less than 5 thes, your will get the cat, for (4) secs
say You'have'too'many'tries'to'get'a'cat.'But'you'can'always'play again for (4) secs
[Основна версия на дейността]
За да спестим време, можем да използваме основната версия на
сценария. в основна версия са включени всички основни понятия,
други фупкции, описани по-торе, могат да обдат използвани като





when clicked	
set number of cats v to pick random (1) to (100)	
repeat until (answer) = number_of_cats	
ask Howmany cats do your think I have? and wait	
if (answer) < (number_of_cats)	
say No.'nol/have/more/cats/than/that! for (2) secs	
if answer > (number_of_cats)	
say lihavelessicats for (2) secs	
stop al	
Инструменти и • Цялата деиност в Snap!: https://snap.berkeley.edu/project?user=zapusek&project=ca	t
ресурси за учителя <u>s in a shelter</u>	2
• Lajovic, S. (2011). Scratch. Nauči se programirati in postani	
računalniški maček. Ljubljana: Pasadena.	
• Vorderman, C. (2017). Računalniško programiranje za otroke.	
Ljubljana: MK.	
Инструменти и • Шаблон в Snap!:	
ресурси за <u>https://snap.berkeley.edu/project?user=zapusek&project=cats_in_a</u>	_
учениците <u>shelter template</u>	
 Инструкции (C4G11_InstructionsForStudent_BG.docx 	





СЦЕНАРИИ ЗА НАПРЕДНАЛИ

Учебен сценарий 12 – Улов на здравословна храна

Учебен сценарий	Улов на здравословна храна
Име	
Предишен опит в програмирането	 Добавяне на текст към спрайт Показване и скриване на спрайт Използване на посока на завъртане (point in direction) Използване на случайни числа Използване н апроменливи за изчисляване на точки Използване на цикъл repeat Using forever loop Using conditionals
Очаквани резултати	 Основни очаквани резултати Променливи Условия Цикли Посока на завъртане Случайни числа Специфични очаквани резултати, ориентирани към алгоритмично мислене: Ученикът използва променлива за предотвратяване започването на играта преди момичето да приключи да говори (по избор) Ученикът използва if за проверка (с помощта на променлива) дали храната може да започне да се движи Ученикът използва цикъл (repeat until) за движение на храната, докато точките не са по-малки от 5 Ученикът използва произволно число за избор на брой стъпки Ученикът използва случайно число за преместване в произволно положение Ученикът използва случайно число за преместване в х (произволно положение
Цел, задачи и	(random), y (fixed) position) (по избор) Кратко описание: Момичето лови храна. Тя трябва да бъде





кратко описание на	внимателна, само здравословни елементи носят точки!
дейностите	Задачи: Учениците трябва да програмират два различни
	спрайта, момиче, което дава инструкции, казва какво да
	прави, за да започне играта и брои точки; и храна, която
	случайно пада от горната част на екрана.
	Освен това учениците могат да добавят променлива и ако
	изявление за предотвратяване движението на храната, преди
	момичето да спре да говори.
	Цел: Учениците ще се научат как да се движат произволно за
	Х стъпки и да изберат позиция, както и как да използват
	променливи и условия за предотвратяване на други
	събития.
Продължителност	45 минути
Стратегия и методи	Активно обучение, обучение, основано на игрови дизайн,
на обучение	решаване на проблеми
Форми на обучение	Индивидуална работа / Работа по двойки
Резюме на	(Мотивация-Въведение, Прилагане, Рефлексия и оценка)
обучението	Момичето лови храна. Всяка здравословна храна носи 1
	точка, докато всяка нездравословна храна отнема 1 точка.
	Играта започва с някои инструкции, дадени от момичето.
	Тогава тя изчезва и храната се появява. Когато играчът събере
	5 точки, храната изчезва и момичето се появява отново.





[Стъпка 1]

Тази дейност е предназначена като индивидуална работа или работа по двойки. Учителят дава някои подсказки, обяснява някои по-трудни части и помага при нужда.

Първоначално се дава на учениците:

- Фон
- Момиче-спрайт

Учениците избират фон и добавят основен спрайт, напр. момиче. Момичето дава някои инструкции в началото и след това се скрива. Както видяхме от предишни дейности, добре е да използваме блок *show*, когато щракнем върху знамето (при повторно възпроизвеждане, ако спрайтът остане скрит). Кодът е например:



Ще се върнем към този спрайт по-късно. Нека сега напишем код за плод.

[Стъпка 2]

Учениците добавят нов спрайт, здравословна храна, напр. ябълка. Първо, те програмират движението на спрайта, което е отгоре надолу, така че те избират следните блокове:



Ако не искат ябълката им да е обърната, те могат да изберат третата опция да не се върти в посока(don't rotate in direction).







За да направите играта по-интересна, броят на стъпките може да бъде избран произволно, така че скоростта не винаги ще бъде еднаква. Например:

move pick random (1) to (2) steps

Следващата стъпка е да помислите какво се случва, когато ябълката дойде в долната част на екрана? В този случай учениците могат да използват блок *touching edge (докосва ръб)* в комбинация с оператора *if*. Ако ябълката докосне ръба, тя ще бъде преместена в някакво произволно положение. Блокове за движение ни предлага следващия блок:

go to random position -

Тази команда ще избира произволно х всякакви у координати и ябълката може да се появи навсякъде на екрана (вижте червените точки на снимката)

Ако искаме ябълката да се показва винаги в горната част на екрана, у стойността може да бъде фиксирана и само х стойността ще бъде



избрана произволно. Със следния код ябълката винаги ще се показва в горната част на екрана (вижте червените точки на снимката).

go to x: pick random (-200) to (200) y: (150)

[Стъпка 3]

Учениците вече могат да направят променлива, точки, която ще използват за броене. Точките трябва да имат начална

set points * to D

стойност 0 (спрайта на момичето).

[Стъпка 4]

Ако искаме ябълката да се движи многократно, имаме нужда от цикъл. Учениците могат да използват цикъл *repeat until u*





да поставят условие. Например те искат играта да завърши, когато достигнат 5 точки. Така че състоянието ще бъде *points* = 5 и цикълът ще се повтаря, докато условието не е вярно. Когато условието е вярно, това е, че играчът достига 5 точки, цикълът ще спре.

[Стъпка 5]

Не искаме ябълката да се показва в началото, но след като момичето даде указанията си. Учениците могат да програмират ябълката да се показва *when key is pressed.* Разбира се, те трябваше да добавят блок *show* преди цикъла repeat и nd *hide* след това. Целият код е:



[Стъпка 6]

Какво се случва, когато ябълката бъде щракната (или въведена с мишката)? Ябълката трябва да се скрие, да преброи точки, да смени

позицията си и да се покаже отново. Точките ще бъдат променени с 1 и за позицията учениците могат да използват същия код като преди.







[Стъпка 7]

Да се върнем при момичето.

Сега момичето трябва да се появи отново и да каже, напр. Поздравления!

Ще се нуждаем от цикъл завинаги, който ще провери дали сме достигнали 5 точки. Ако го направиме, момичето ще се покаже и ще каже нещо. След това ще добавим блокиране на всички. Нека учениците да разберат какво означава това спиране (без спиране момичето ще казва "Поздравления …" завинаги).



[Стъпка 8]

Когато играете отново играта, когато учениците вече ще знаят всички инструкции (от [Стъпка 1]) и със сигурност ще искат да ги пропуснат. Те могат да натиснат буквата "S" преди, така че играта да започне, но момичето все още ще говори. За да предотвратим това, можем да създадем друга променлива (с име start), която трябва да бъде зададена на 0 в началото. След това, след инструкциите на момичето, променливата старт ще се промени на 1.







променливата start е равна на 1. Учениците ще използват блока *if*. С това учениците няма да могат да провеждат игра, преди момичето да спре да говори.

Друго нещо може да се случи, когато играем отново играта. Ако спрем играта, когато имаме например 3 точки, ябълката няма да изчезне. В този случай, когато започнете играта отново, ябълката ще бъде видима, преди момичето да завърши с даване на инструкции. Тъй като не искаме това, ние добавяме код, който apple скрива в началото на играта.

Кодът на ябълката вече е:



[Стъпка 9]

Сега учениците могат да дублират ябълковия спрайт много пъти и да им сменят костюма (ако искат). Кодът ще бъде същият.

Единствената промяна е при нездравословната храна, където те ще загубят 1 точка, като щракнат върху нея.

change points by 1

[Финален код]





Girl/Момиче when 🍋 clicked set points to 0 set starie to 0 show say Hello! for </u> secs say Help-me-to-catch-the-healthy-food! for (4) secs say Healthy-food-brings-1-point,-unhealthy--1. for (4) secs say The-game-ends-when-you-reach-5-points, for (4) secs say Press-S-to-start-the-game! for (2) secs hide set start to 1 forever if points = 5 show say Congratulations! You have collected enough healthy food! for (5) secs stop al-Apple/ябълка when s * key pres en clicked if start = 1 points by 1 show go to x: pick random 200 to 200 y: 150 repeat until 15 = points point in direction 180 move pick random (1) to (2) steps touching edge 7 go to x: pick random _200 to 200 y: 150 hide [Допълнителни задачи] Студентите могат да добавят допълнителни задачи според техните желания или могат да следват задачите по-долу: • Променете играта, така че спрайтът на купата да улавя храна. • Добавете нов спрайт (купа). Начертайте го, намерете го онлайн или използвайте приложена снимка / и на купата. Задайте началната позиция на купата (например в долната част на екрана) и напишете код за движението





	на купата (наляво и надясно, ако искате и нагоре и
	надолу). Хранителните спрайтове трябва да изчезнат и
	да се появят отново на произволно място чрез
	докосване на купата (а не при щракване с мишката
	върху храната, както преди).
	• Променете правилата - оставете играта да приключи,
	когато играчът набере 20 точки (той печели) или когато
	вземе 3 нездравословни храни (загуби).
	• Добавете още хранителни спрайтове, за да направите
	играта по-интересна.
	• Сменете костюма на купата, когато играч отбележи
	напр. 5, 10, 15 точки.
Инструменти и	 Цялата дейност в Snap!:
ресурси за учителя	https://snap.berkeley.edu/project?user=mateja&project=
	Catching%20healthy%20food
	 Дейност в Snap! С допълнителни задачи:
	https://snap.berkeley.edu/project?user=mateja&project=
	<u>https://snap.berkeley.edu/project?user=mateja&project=</u> <u>Catching%20healthy%20food%20%2B%20Add.%20Task</u>
	 <u>https://snap.berkeley.edu/project?user=mateja&project=</u> <u>Catching%20healthy%20food%20%2B%20Add.%20Task</u> Lajovic, S. (2011). Scratch. <i>Nauči se programirati in postan</i>
	 <u>https://snap.berkeley.edu/project?user=mateja&project=</u> <u>Catching%20healthy%20food%20%2B%20Add.%20Task</u> Lajovic, S. (2011). Scratch. <i>Nauči se programirati in postan</i> <i>računalniški maček</i>. Ljubljana: Pasadena.
	 <u>https://snap.berkeley.edu/project?user=mateja&project=</u> <u>Catching%20healthy%20food%20%2B%20Add.%20Task</u> Lajovic, S. (2011). Scratch. <i>Nauči se programirati in postan</i> <i>računalniški maček</i>. Ljubljana: Pasadena. Vorderman, C. (2017). <i>Računalniško programiranje za</i>
	 <u>https://snap.berkeley.edu/project?user=mateja&project=</u> <u>Catching%20healthy%20food%20%2B%20Add.%20Task</u> Lajovic, S. (2011). Scratch. <i>Nauči se programirati in postan</i> <i>računalniški maček</i>. Ljubljana: Pasadena. Vorderman, C. (2017). <i>Računalniško programiranje za</i> <i>otroke</i>. Ljubljana: MK.
Инструменти и	 https://snap.berkeley.edu/project?user=mateja&project= Catching%20healthy%20food%20%2B%20Add.%20Task Lajovic, S. (2011). Scratch. Nauči se programirati in postan računalniški maček. Ljubljana: Pasadena. Vorderman, C. (2017). Računalniško programiranje za otroke. Ljubljana: МК. Полуготови шаблони в Snap!:
Инструменти и ресурси за	 https://snap.berkeley.edu/project?user=mateja&project= Catching%20healthy%20food%20%2B%20Add.%20Task Lajovic, S. (2011). Scratch. Nauči se programirati in postan računalniški maček. Ljubljana: Pasadena. Vorderman, C. (2017). Računalniško programiranje za otroke. Ljubljana: МК. Полуготови шаблони в Snap!: https://snap.berkeley.edu/project?user=mateja&project=
Инструменти и ресурси за учениците	 https://snap.berkeley.edu/project?user=mateja&project= Catching%20healthy%20food%20%2B%20Add.%20Task Lajovic, S. (2011). Scratch. Nauči se programirati in postan računalniški maček. Ljubljana: Pasadena. Vorderman, C. (2017). Računalniško programiranje za otroke. Ljubljana: MK. Полуготови шаблони в Snap!: https://snap.berkeley.edu/project?user=mateja&project= C4G12_Catching%20healthy%20food%20-%20Part
Инструменти и ресурси за учениците	 https://snap.berkeley.edu/project?user=mateja&project= Catching%20healthy%20food%20%2B%20Add.%20Task Lajovic, S. (2011). Scratch. Nauči se programirati in postan računalniški maček. Ljubljana: Pasadena. Vorderman, C. (2017). Računalniško programiranje za otroke. Ljubljana: MK. Полуготови шаблони в Snap!: https://snap.berkeley.edu/project?user=mateja&project= C4G12_Catching%20healthy%20food%20-%20Part Инструкции (C4G12_InstructionsForStudent_BG.docx)





Учебен сценарий 13 – Разказване на история

Учебен сценарий	Разказване на история
Име	
Предишен опит в	 Показване и скриване на спрайт
	 Използване на посока на движение (Point in direction)
програмирането	 Използване на условни изрази и условен блок
	 Използване на блок Кажи
	 Използване на блок Изчакай за секунди
Резултати от	Общи резултати от обучението:
обучението	 Основни очаквани резултати:
	 Преместване и промяна на размера на спрайт
	• Промяна на размера
	• Съобщения
	 Съставете структура на разказването на истории
	 Промяна на фона на сцените
	Специфични резултати от обучението, ориентирани към
	алгоритмично мислене:
	• Ученикът планира диалози и дейности на спрайтовете в
	историята
	• Ученикът изпращане на предавания за диалог между
	спрайтове
	• Ученикът използва движещи се и променящи се размери за
	спрайтове
	 Ученикът използва показване и скриване на спрайта
	 Ученически рефактор и разширен код на спрайтовете
Цели, Задачи и	Кратко описание: Заекът разказва историята за Алиса в страната на
кратко описание	чудесата. Той започва разказа на историята с няколко изречения в
на дейностите	началото на фона на декор на сцена с надпис Allice in Wonderland.
	Историята на Алиса започва в гората. Алиса се разхожда и си мисли:
	"Къде съм?" /За да се реализира отдалечаването на Алиса,
	постепенно с придвижването ѝ се намалява размера. /Алиса попада
	на кръстопът и на дърво вижда Чеширския котарак/. Започва
	разговор между Алиса и Чеширския котарак.
	Разговорът е представен на картинката.
	Задачи: Учениците трябва да експериментират с кратък пример на
	историята за срещата между Алиса и котарака, базирана на





	синхронизиране на диалога чрез блок за изчакване. След това
	разглеждат втора версия на историята с използване на съобщения.
	Въвеждат се команди за предаване на съобщения. учениците
	допълват кода на героите по текста от картинката. Задачата се
	усложнява с въвеждане смяна на декор на сцена чрез broadcast и
	движение на Алиса в гората преди да срещне котаракът.
	Цел: Учениците ще се научат как да планират разказване на
	истории, как да използват излъчени съобщения за синхронизиране
	на дейностите на спрайтовете и сценични промени.
Продължителност	90 минути
Методи на	Активно учене, обучение, основано на програмиране на игра,
обучение	решаване на проблеми
Форма на	Самостоятелна работа / Работа по двойки
преподаване	
Ход на урока	(Мотивация-Въведение, Прилагане, Осмисляне и Оценка)
	Учителят дискутира с учениците историята на Алиса в страната на
	чудесата и показва картинката със срещата на Алиса и чеширския
	котарак. Обяснява, че с помощта на кодирането може да се
	пресъздаде историята на Алиса. Поставя задача да се стартира
	проекта и да се разгледат кодовете на sprites
	https://snap.berkeley.edu/project?user=ddureva&project=Alice 1
	Дискутира се: Кой започва да говори първи? Кога се включва Алиса и
	кога Котаракът? Защо при диалога на героите няма синхрон?
	Отговорът е в некоректното изчисляване на времената в които всеки
	от героите "говори" и "неизчакването даден герой да приключи
	репликите си".
	Rabit
	Surjet Commen Scorols
	say Hittig for CD secs say Hittig for CD secs say Hittig for CD secs say Hittig for CD secs





Заек Кажи: ЗдравейтеЧували ли сте за Алиса инейните приключения в Страната на чудесата? Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=0 Кои са непознатите до момента команда	стартиране от началото 0 9 10 10	за край 14 21 20	жител- ност 14 12 10
Заек Кажи: ЗдравейтеЧували ли сте за Алиса инейните приключения в Страната на чудесата? Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на геристории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=0 Кои са непознатите до момента команд	от началото 0 9 10 лок wait for	край 14 21 20	ност 14 12 10
Заек Кажи: ЗдравейтеЧували ли сте за Алиса инейните приключения в Страната на чудесата? Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=0 Кои са непознатите до момента команд	началото 0 9 10 лок wait for	14 21 20	14 12 10
Заек Кажи: ЗдравейтеЧували ли сте за Алиса инейните приключения в Страната на чудесата? Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=0 Кои са непознатите до момента команд	0 9 10 лок wait for	14 21 20	14 12 10
ЗдравейтеЧували ли сте за Алиса инейните приключения в Страната на чудесата? Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=4 Кои са непознатите до момента команд	9 10 лок wait for	21 20	12
сте за Алиса инейните приключения в Страната на чудесата? Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=4	9 10 лок wait for	21	12
приключения в Страната на чудесата? Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=4	9 10 лок wait for	21 20	12
Страната на чудесата? Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=4	9 10 лок wait for	21 20	12 10
Нека да видим една от нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=e Кои са непознатите до момента команд	9 10 лок wait for	21 20	12 10
нените истории. Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=4 Кои са непознатите до момента команд	9 10 лок wait for	21 20	12 10
Алиса Казва: Би ли ми казал по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти <u>https://snap.berkeley.edu/project?user=c</u> Кои са непознатите до момента команд	9 10 лок wait for	21	12 10
по кой път да тръгна? Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. L. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=0 Кои са непознатите до момента команд	10 лок wait for	20	10
Котаракът Казва: А ти къде искаш да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти https://snap.berkeley.edu/project?user=0 Кои са непознатите до момента команд	10 лок wait for	20	10
да отудеш? Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти <u>https://snap.berkeley.edu/project?user=</u> Кои са непознатите до момента команд	лок wait for		
Изводът е, че синхронизирането чрез б доведе до грешки в поведението на гер истории. 1. Учителят поставя задача да се старти <u>https://snap.berkeley.edu/project?user=c</u> Кои са непознатите до момента команд	лок wait for		1
доведе до грешки в поведението на геј истории. 1. Учителят поставя задача да се старти <u>https://snap.berkeley.edu/project?user=(</u> Кои са непознатите до момента команд		second	може д
истории. 1. Учителят поставя задача да се старти <u>https://snap.berkeley.edu/project?user=e</u> Кои са непознатите до момента команд	роите при ра	зказване	е на
 Учителят поставя задача да се старті <u>https://snap.berkeley.edu/project?user=c</u> Кои са непознатите до момента команд 			
https://snap.berkeley.edu/project?user= Кои са непознатите до момента команд	ира и разгле	да кода н	на прое
Кои са непознатите до момента команд	ddureva&pro	ject=Alice	<u>e 2</u>
	ци?		
Сравняват се кодовете от Alice_1 и Alice	e_2.		
O Alice 1			
Edf 2 dragatite	Alice 2 draggadia		
South Contanes Sounds South South	Costanes Sounds		
au to z: •74) y: •11) show well 9	187.) v: 6 67.)		
well \$20 sectors a good dear on WHERE you want to control for \$10 sectors WHERE you want to control for \$10 sectors WHERE		ri ouahtto ao fram h	terel for (10)





Alice_1	Alice_2
Alice 1	Altes 2 Additional and a second se
Alce V angustik Vangustik Verber Coheres Bundle velen Chiter velen	Alce 2 Alce : Alce :
Alice 1 Gal	Alice 2 Cat Support Support Cat Suport Cat Suport Cat Support
2. Въвеждат се блоковете за съобще	ения:
broadcast 💌 broadcast 💌 and wait	when I receive
Коментира се, че съобщенията зада	дени с Broadcast са насочени към
всички герои, но могат да се полу	чават само от някои от героите.
Блокът broadcast and wait из	исква всички герои получили
съобщението да изпълнят действ	ията си и тогава продължават
действията на героя изпратил съоб	бщението. Учителят демонстрира
как се задава име на съобщение в	broadcast и как се използва при
събитието When I receive	
1. Metrage r	ансм 3. Въвеждане на име. ОК











Коментират се смяната на сцените и действията на героите. Кога се сменя дадена сцена, кога се появява Alice и какви са нейните действия? Кога се появява котаракът и какви са неговите действия?

Разглеждат се сцените в проекта Alice_2. Има 3 сцени, едната вече е използвана. Коя сцена да се използва за начало. Какво трябва да се направи, така че при стартиране на историята Alica I Cat да не се виждат? Как да сменим костюма на сцената? Може да се използва broadcast за смяна на сцената зададен от Заека, след като каже въвеждащите думи в историята. Алиса се появява при смяна на сцената със съобщението Go to forest







	Когато достигне кръстопътя се сменя сцената със съобщението
	"Meeting with Cheshire Cat". Това съобщение се получава
	едновременно и от Заека, който намалява размера си на 80% и
	продължава да разказва историята. На този етап котаракът не се
	показва,защото присъства като част от декора. Той се появява при
	съобщението Cat1. Учителят може да обясни, че котаракът е изрязан
	от декора с помощта на външен графичен редактор. (За съжаление
	Snap! не предоставя големи възможности на графичния редактор за
	разлика от Scratch 3.0).
	След съобщението на Заека Alice 1 историята продължава, така както
	е направена в проекта Alice 2.
	7. Учителят коментира, че за да се разкаже една история най-напред
	трябва да се направи нейния сценарии. За целта може да се
	използва допълнителна таблица за описание на сценария на
	историята. (Приложение 1) По преценка на учителя може да се
	даде готовата таблица или да се даде частично попълнена, а
	учениците базирайки се на демонстрацията да я попълнят.
	8. Поставя се задачата учениците да опишат разгледания сценарии и
	да довършат историята от проекта Alice_2 по двойки.
Ресурси за	Цялата дейност е в Snap!:
учителя	https://snap.berkeley.edu/project?user=ddureva&project=Alice
Ресурси за	
учениците	https://snap.berkeley.edu/project?user=ddureva&pro
	ject=Alice 1
	https://snap.berkeley.edu/project?user=ddureva&pro
	iect=Alice 2
	Инструкции за ученици
	(C4G13 InstructionsForStudent BG.docx)





Сцени

Име	Дизайн	Десйтвие	Бележки
1. Start	Alice in the Wonderland	Историята	На този фон Заекът прави
		започва със	въведение в историята.
		сцената. (When	
		green flag is	
		clicked)	
2. Forest		Фонът на	На този фон Алиса се появява
		сцената се	позиционирана в центъра на
		появява когато	сцената. Започва да се движи
		заекът	мислейки "Къде съм?".
		приключи с	Постепенно намалява 5 пъти
		въведението в	размера си с 10% и мисли.
		историята.	Когато стига в края на пътя (до
		(Изпратил е	един кръстопът) сцената се
		съобщение Go	сменя с декор Meeting. (Алиса
		to forest)	дава съобщение Meeting with
			Cheshier Cat)
3. Meeting		Появява се	Тук Алиса и котаракът са част
		когато получи	от фона. За да се използва
		съобщението от	спрайта на Алиса, преди
		Алиса Meeting	съобщението, тя се
		with Cheshier Cat	позиционира, така че да
			покрие изображението и от
			декора. Спрайта на котаракът
			се появява на по-късен етап.
			Със смяната на сцената Заекът
			продължава да разказва
			историята.
			По-късно се провежда
			разговорът между алиса и
			Чеширския котарак.







Герои (Спрайтове)

Спрайт	Действие	Фон на сцена
Rabbit	При старт: Казва: Здравей! за 2 сек. Казва: Чувал ли си някога за Алиса и нейните приключения в Страната на чудесата? за 6 сек. Казва: Сега нека да видим нейната история! за 6 сек	in the Wandschaat
	Изпраща съобщение Go to forest /Иди в гората	
Alice	При старт Скрива се от сцената. Позиционира се в центъра на сцената и се задава размер 100% в готовност за показване в новия фон.	in ha Wanderhaat Start
Cat	При старт Скрива се от сцената. Позиционира се в x: -74, y:113 (Позициите са предварително определени след наместване на спрайта на котарака върху сцената Meeting.)	in he Winderhard Start
Alice	Получава съобщение Go to forest: Показва се 5 пъти се повтаря: изчакване за 1 сек.; преместване с 5 стъпки; намаляване на размера (смяна с -10); "Мисли" – "Къде съм?" Подготвя се за следващия декор: Изчаква 5 сек; Възстановява размера си (смяна 100%) и се позиционира в х:-187, у:-67 Изпраща съобщение: Meeting with Cheshiere Cat/ Среща с котарака.	forest
Rabbit	Няма действие. Само е показан от предходния декор.	forest
Rabbit	Получава съобщение: Meeting with Cheshiere Cat. Сменя размера си на 80% Казва: "Алиса спряла н акръстопътя и се чудела на къде да	meeting





	тръгне." за 10 сек.	
	Казва: "Тя видяла Чеширския котарак на дървото." за 8 сек.	
	Изпраща съобщение Alice1	
Carl	Получава съобщение: Alice1.	- CART
Alice	Премества се на преден слой (Това е необходимо, защото	14 20
10102	след нея се появява котаракът, който пречи да се	meeting
	визуализират репликите на Алиса, ако тя не е в предния	
	слой).	
	Казва: "Ні!" за 2 сек.	
	Казва: "Would you tell me please, which way I ought to go from	
	here!" за 10 сек.	
	Изпраща съобщение (broadcast) към Котаракът: Cat1.	
	Получава съобщението Cat1.	200
Cat	Показва се.	100
	Казва: "А ти къде искаш да отидеш?" за 10 сек.	meeting
	Изпраща съобщение "Alice2"	
	Получава съобщение: Alice 2.	200
Alice	Казва:	14 Cash
	Изпраща съобщение Cat2	meeting
3	Получава съобщение Cat2.	200
Cat	Казва:	1.20
Gat	Изпраща съобщение: Rabbit1	meeting
and the	Получава съобщение Rabbit1	KANETOPPE
	Казва: "Каква е поуката от тази история?" за 8 сек.	WT
Rabbit	Казва: "За да знаеш по кой път да тръгнеш. трябва да	
	определиш каква е целта ти."	meeting
	- Leffer	





Учебен сценарий 14 – Рисуване

Учебен сценарий	Рисуване
Име	
Предишен опит в	Добавяне на спрайт
програмирането	 Задаване на посока на завъратнае (point in direction) Използване на променливи за изчисляване на точки Използване на цикъл Използване на разклонения (условия)
Очаквани	Осовни очаквани резултати:
резултати	 Променливи Разклонения и условия Цикъл Посока на джижение оператори
	Специфични очаквани резултати, ориентирани към
	алгоритмично мислене:
	 ученикът използва писалка за рисуване ученикът използва цикли за рисуване ученикът променя стойността на променлива при рисуване ученикът използва точка в посока, за да рисува предмети на сцената ученикът използва излъчване за управление на спрайт ученикът използва условни условия за смяна на сцената ученикът използва оператор> за смяна на сцената
Цел, задачи и	Кратко описание: Климатът се е променил много, въздухът е
кратко описание	силно замърсен поради индустрията. Трябва да се засаждат
на дейностите	дървета, за да се подобри качеството на въздуха!
	Задачи: За да подобрят качеството на въздуха, учениците трябва
	да програмират спрайт, да нарисуват два вида различни дървета
	- бор и дъб и бутони, които символизират тези видове дървета.
	Когато се натисне бутон, се изчертава определен тип дърво.
	Цел: Учениците ще се научат да рисуват в Snap !, за да
	променят цвета и дебелината на писалката и как да използват
	променливи и условни условия, които причиняват ново
	събитие.





Продължителност	45 минути
Стратегия и	Активно обучение, обучение, основано на игрови дизайн,
методи на	решаване на проблеми
обучение	
Форми на	Индивидуална работа / Работа по двойки
обучение	
Резюме на	(Мотивация-Въведение, Прилагане, Рефлексия и оценка)
обучението	В началото на играта на сцената се показват индустрия, която
	причинява климатични промени и променлива, която показва
	качеството на въздуха. Трябва да се засаждат дървета, за да се
	подобри качеството на въздуха. Могат да се нарисуват два
	различни вида дървета, бор и дъб. Когато е изтеглен бор,
	въздухът се подобрява с 3, а чрез изчертаване на дъб въздухът се
	подобрява с 2 единици. Когато качеството на въздуха достигне
	10 единици, сцената се променя на поляна.
	[Стъпка 1]
	Учениците трябва да отворят програмата за подобряване на
	климата, която съдържа шаблон на фонове (промишленост и
	трева) и спрайтове (молив, бор, дъб и спрайт, наречени ясно).
	Също така, добавете нов спрайт - pencil ("pencil a" от
	предложените спрайтове). Тъй като спрайтът е твърде голям, той
	трябва да бъде намален до 50%. Трябва да се посочи и
	началната позиция на молива (координатите), напр. Х = -10, у = -
	10.
	when clicked show set size to 50 % go to x: -10 y: -10 when clicked switch to costume industry
	[Стъпка 2]





Спрайтът с молив трябва да получава съобщения "дъб" и "бор" и да изчертава подходящи дървета в отговор на съобщението. Първо маркирайте спрайта на молива и добавете кода, който ще позволи рисуването на бор с помощта на молив, когато спрайтът получи съобщението "бор". Посоката на завъртане трябва да бъде зададена на 90, за да се очертае короната във формата на триъгълник и да се зададе цветът му. in direction 90 -За да нарисувате цялата корона, преместете спрайта на 40 step 40 (120) degree стъпки, завъртете наляво на 120 градуса. at 3 Това движение трябва да се повтори три пъти. След короната, стеблото също трябва да бъде изчертано. За да бъде стеблото в правилното положение, преместете го с 22 стъпки. 22 set pen color to След това задайте кафяв цвят на молива. Завъртете на 90 градуса надясно и след това преместете 10 turn (2 90) degrees move 10 steps стъпки. Това движение трябва да се повтори 3 пъти. В крайна сметка е необходимо да вдигнете молива нагоре, така че спрайтът да не остави следа по време на следващото движение. Също така моливът трябва да се премести в произволно положение.





repeat (120

go to x: pick random (210) to (220) yr pick random (30) to (160)

[Стъпка 3]

По същия начин е необходимо да добавите кода към молив спрайт за рисуване на дъбове. Дъбът трябва да бъде нарисуван, когато спрайтът получи съобщението "дъб". Посоката на въртене в посока трябва да бъде настроена на 90, за да се запази короната в кръг, писалката трябва да е надолу и да се зададе цвят.



За да нарисувате короната на дъба, преместете спрайта на 1 стъпка и завъртете 3 градуса наляво след всяка стъпка.

move 1 steps turn 5 3 degrees

Това движение трябва да се повтори 120 пъти.

След като нарисуват екороната, трябва да изчертаете ствола на дървото. Спрайтът молив трябва да се премести в центъра на нарисувания кръг с -3 стъпки и цветът на писалката да се промени на кафяв.

> move 🖪 steps set pen color to

За да нарисува ствола на дъба, спрайтът трябва да се обърне надясно на 90 градуса и да се премести с 10 стъпки.

turn 👌 90 degrees move 10 steps

Тази част се повтаря три пъти.






Когато рисуването е готово, е необходимо да вдигнете молива нагоре, така че да не очертава линията, когато спрайта на молива се движи.



След изрисуването на дъба, моливът трябва да се премести в произволна позиция.

go to z: pick random (210) to (220) y: pick random (30) to (180)

[Стъпка 4]

Ученици трябва да добавят кода, който кара всички изчертани дървета да бъдат изтрити, когато играчът щракне върху спрайта за изчистване Clear. Когато щракнете върху спрайта Clear с мишката, той разпространява (изпраща) съобщение за изчистване на всички дървета. Когато спрайтът Pencil получи съобщение, той изтрива всички изчертани дървета.

Subara I success division
Statement of the second s
clear

[Стъпка 5]

Създайте нова променлива "clean air", за да покажете текущото качество на въздуха. Задайте началната стойност 0 и покажете променливата на сцената.

set clean air to 🛛

Всеки път, когато се изчертава бор, въздухът се подобрява с 2 единици, така че добавете блок към спрайта бор (pine), който ще промени стойността на променливата "clear air) с 2 всеки път, когато щракнете върху бор.







Всеки път, когато се изчертае дъб, въздухът се подобрява с 3 единици, така че добавете блок към спрайта дъб (oak), който ще промени стойността на променливата "clear air" с 3 всеки път, когато щракнете върху дъб.



[Стъпка 6]

Когато променливата "clear air" достигне 10, сцената трябва да се промени на трева. Затова от изтеглените материали добавете нов фон "grass" за сцената (фонът е от изтеглените материали,).



Добавете блок "When" от групата "Control" към спрайта молив.



След това добавете оператор за сравнение >.



Задайте на спрайта да разпространи съобщение "grass"когато променливата "clean air" стане по-голяма от 10.



Добавете кода на сцената, за да смените костюма на "grass", когато се получи съобщението "grass".







	[ДОПЪЛНИТЕЛНА ЗАДАЧА]
	Можете да надстроите играта, като добавите животни, които се
	появяват, когато въздухът вече не е замърсен.
	[Краен код]
	Pine
	when I am clicked a broadcast draw a pine a
	Oak
	weheen & arm clicked # broadfcast draw an calk #
	x
	when I am clicked - broadcast clean -
	Pencil
	when i takket when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part when i preserve take part <t< th=""></t<>
	Stage
	when I receive grass switch to costume industry
Инструменти и	Snap! проект "Drawing":
ресурси за учителя	https://snap.berkeley.edu/project?user=tadeja&project=Improve%20the% 20climate (9.1.2020)
Инструменти и	 Programming language Snap!: <u>https://snap.berkeley.edu/</u>
ресурси за	(9.1.2020)
учениците	 инструкции за ученици (C4G14 InstructionsForStudent BG.docx)









Учебен сценарий	Хвани мишката	
Име		
Предишен опит в програмирането	 Ученикът добавя фон. Ученикът добавя нов спрайт. Ученикът добавя нов звук. Ученикът кара спрайт да каже нещо. Ученикът променя костюма на спрайта, за да направи анимация. Ученикът е реализира движението на обекта с клавишите със стрелки, използвайки събития и взема предвид ограниченията Ученикът прави разлика между две различни състояния и знае как да ги изрази с логически изрази. Ученикът използва условни условия. 	
Очаквани	Основни очаквани резултати:	
резултати	 Цикъл forever; Случайни числа; Брояч; Таймер. Специфични очаквани резултати, ориентирани към алгоритмичното 	
	мислене:	
	 ученикът използва цикъл forever за преместване на спрайтовете; ученикът използва случайни числа, за да определи позицията на спрайта, премества спрайта за произволни стъпки и завърта спрайта за случайни градуси; ученикът прилага брояч за отчитане на улова на мишки и използва крайната стойност, за да обобщи доколко е успешен играчът; ученикът използва таймер, за да определи края на играта. 	
Цел, задачи и	Кратко описание: Програмирайте игра, в която играчът (котката) ще	
кратко описание	трябва да хване мишката.	
на дейностите	Задача: Програмирайте дейността, при която котката ще хване мишката.	
	Котката ще бъде преместена от играча с клавиши със стрелки и мишката	
	ще се движи произволно. Когато котката докосне мишката, мишката ще	
	се скрие и ще се появи на произволно място. Трябва също да имаме	
	брояч, който да отчита колко пъти котката е хванала мишката. Трябва	
	също да имаме от таймер, за да завършим играта. След дейността	
	момичето трябва да обобщи колко успешен играч е, тя казва колко пъти	





	играчът е хванал мишката.	
	Цел: Ученикът ще се запознае с концепцията за множество	
	променливи случайни стойности. Те ще се научат как да използват	
	Operators/pick random[x]to[y] block.	
Продължителност	45 минути	
Стратегия и	Активно обучение, съвместно обучение, решаване на проблеми,	
методи на	обучение, основано на игрови дизайн	
обучение		
Форми на	Фронтално обучение	
обучение	Работа по двойки / групова работа	
Резюме на	(Мотивация-Въведение, Прилагане, Рефлексия и оценка)	
обучението	Мотивация-Въведение	
	Мотивираме учениците, като показваме играта. Обсъждаме с тях как	
	биха започнали да програмират тази игра. Заедно със учениците	
	определяме последователността на стъпките, например:	
	1. изберете фон и добавете спрайтове;	
	2. програмирайте котката да се движи с клавишите със стрелки;	
	3. програмирайте мишката да се движи произволно;	
	4. програмираите мишката да се скрие (и да се появи на произволно място), когато докосне котката;	
	5. брояч на програми;	
	6. добавете таймер и определете края на играта;	
	7. добавете момичето и я програмирайте, за да обобщите колко	
	8. програмирайте момичето да скача, когато докосне мишката:	
	9. добавете звук на котката / мишката;	
	10. и т.н.	
	Учениците могат да помогнат със стъпките или да създадат свои	
	собствени правила на играта (но трябва да следват смели стъпки).	













[Стъпка 3]

Учениците трябва да програмират мишката да се движи произволно. В този случай идеята е, че мишката в цикъл forever прави произволен брой стъпки и се обръща за произволна степен. Студентите правят това с блоковете *Motion/move[x]steps* и *Motion/turn[x]degrees*, в които тв вмъкват оператора *pick random[x]to[y]* operator.



[Стъпка 4]

Следващата стъпка е да програмирате мишката да се скрие, когато докосне котката. Идеята е, че мишката се скрива и се появява на произволно място, когато докосне котката. В този случай играта не





завършва при първия улов на мишката. Учениците могат да добавят собствено правило тук. Във всеки случай те трябва да използват оператора *pick random[x]to[y]*.



[Стъпка 5]

В случай, че искаме да знаем колко пъти мишката е била уловена, трябва да добавим брояч. Учениците правят нова променлива - score (резултат) и я добавят към кода на котката. Резултатът в началото винаги трябва да е 0. Учениците правят това с Variables/set[variable]to[x] b. Ако искаме резултатът да бъде показан, учениците трябва да добавят блока show variable[variable]. След това учениците добавят нов контролен блок (Control/when) за да проверят дали котката докосва мишката. Ако котката докосне мишката, резултатът ce увеличава С 1 (Variables/change[score]by[x]).







време (напр. 30 секунди) мишката и котката изчезват, променливата Резултат се скрива и играта приключва.

Учениците трябва да добавят тези блокове към скрипта за котка и мишка.

[Стъпка 7]

Учениците трябва да добавят код за момичето, за да обобщят крайния резултат. Ако играчът не хване никакви мишки, момичето казва: "Не сте хванали никакви мишки!". В противен случай тя казва: "Поздравления! Хванахте х мишки! "



[Допълнителни задачи]

Учениците могат да добавят всякакви елементи към играта си.

Например момичето, което скача всеки път, когато докосне мишка.



Учениците могат да добавят звук. Например, те добавят звук на котката. Звукът се възпроизвежда, когато мишката бъде уловена.





when touching Mouse ? change Score by (1) play sound Meow wait (1) secs
Рефлексия и оценка
 мишката се движи 20 до 60 стъпки завинаги; мишката отива на място х = 100, когато докосне котката; мишката се завърта завинаги на 90 градуса; и т.н.
The mouse
when ticked show betwee If touching Cut ? go to x: pick random 200 to 200 y: pick random 200 to 200 move pick random 10 to 100 steps If on edge, bounce wait 0.35 secs turn c pick random 20 to 90 degrees If on edge, bounce
The cat











Инструменти и	 Цялата дейност в Snap!:
ресурси за учителя	https://snap.berkeley.edu/project?user=tadeja&project=Catch%20th
	<u>e%20mouse</u>
	 Безплатни изображения: <u>https://pixabay.com/</u>
	 Lajovic, S. (2011). Scratch. Nauči se programirati in postani računalniški maček. Ljubljana: Pasadena.
	• Vorderman, C. (2017). <i>Računalniško programiranje za otroke</i> .
	Ljubljana: MK.
Инструменти и	• Шаблон в Snap!:
ресурси за	https://snap.berkeley.edu/project?user=tadeja&project=Catch%20th
учениците	<u>e%20mouse_0</u>
	 Безплатни изображения: <u>https://pixabay.com/</u>
	 Инструкции (C4G15_InstructionsForStudent_BG.docx)





Учебен сценарий 16 – Купуване на храна за пикник

Учебен сценарий	Купуване на храна за пикник
Име	
Предишен опит в	• Добавяне на текст към спрайт
програмирането	 Показване и скриване на спрайтове
	• Използване на оператори
	• Използване на променливи
	• Използване на слепване на стринтове
Очаквани резултати	Основни очаквани резултати:
	 Променливи Условни блокове Оператори Специфични очаквани резултати, ориентирани към алгоритмично
	мислене:
	 Ученикът използва променливи за определяне на цената за различни спрайтове - храни Ученикът променя стойността на променливите, тъй като бюджетът се променя, когато играчът купи храна Ученикът използва if оператор за проверка на наличността на пари Ученикът използва оператори за присъединяване на текст - стойност на променливите - текст Ученикът използва оператори за сравняване на цени и бюджет Ученикът използва оператори (изваждане) за промяна на стойността на променливите
Цел, задачи и	Кратко описание: Момичето отива на пикник и се нуждае от
кратко описание на	помощ при закупуване на храна. Тя има 15 EUR и не може да
дейностите	похарчи повече. Когато тя купи нещо, стойността на бюджета се
	променя. Ако бюджетът ѝ е твърде нисък, тя не може да купи
	избраната храна.
	Задачи: Учениците трябва да програмират три различни спрайта:
	момиче, храна (която могат да дублират с леки промени) и бутон
	за завършване. Момиче дава инструкции, казва колко пари има
	играчът и накрая (с щракване върху бутона за финал) тя казва
	колко здравословни и нездравословни продукти е купил играчът.





	Храната казва цената си, когато курсорът на мишката я задържи.
	Ако играчът има достатъчно пари, той може да закупи продукт и
	стойността на бюджета се променя. В противен случай храната не
	може да бъде закупена.
	Цел: Учениците ще се научат как да работят с променливи:
	задаване на различни начални стойности, използване на условни
	условия за сравняване на стойността на променливите, промяна
	на стойността на променливите, използване на променливи за
	отчитане на (не) здравословна храна. Освен това те ще повтарят
	добавянето на текст, обединяването на текстове и ако
	изявление.
Продължителност	45 минути
Стратегия и методи	Активно обучение, обучение, основано на дизайн на игри,
на обучение	решаване на проблеми
Форми на обучение	Индивидуална работа / Работа по двойки
Резюме на	(Мотивация-Въведение, Прилагане, Рефлексия и оценка)
обучението	Момичето е в бакалия и купува храна за пикник. Тя има 15 EUR. Тя
	може да види цената на храната, когато се задържи курсорът на
	мишката и да я купи, като се щракне върху избраната храна. Тя
	може да купува храна само докато има достатъчно пари. Купете,
	като щракнете върху бутона за завършване, тя казва колко
	здравословни и нездравословни продукти е купил играчът.
	Cake costs 8 EUR.





[Стъпка 1]

Тази дейност е предназначена като индивидуална работа или

работа по двойки. Учителят дава някои улики, обяснява някои по-

трудни части и помага при нужда.

Учениците избират фон и добавят основен спрайт, напр. момиче. Момичето дава някои инструкции в началото, напр .:

> say Hello! for (2) secs say linavera-picnic-today-help-me-to-buy-some-food! for (4) secs

[Стъпка 2]

В тази игра ще ни трябват няколко променливи:

- *budget,* за определяне на размера на наличните пари,
- healthy_food, за преброяване колко здравословни елементи е купил играчът,
- *unhealthy_food*, за преброяване колко нездравословни елементи е купил играчът,
- a variable for each food, e.g. watermelon_price, за определяне на цената на всяка храна.

В началото променливата *budget* има стойност например 15 (EUR).

Другите две променливи са зададени на 0. Този код може да бъде добавен преди кода на момичето от [Стъпка 1].



[Стъпка 3]

Учениците добавят спрайт (храна) и избират костюма му.

Кодът на храната (динята) се нуждае от три контролни събития:

 a) When the green flag clicked: да покаже и определи цената на храната. Нека цената на променливата бъде разумно определена (разбира се, не 0, а по-голяма от 1).





stow set watermelon_price to E
b) When mouse-entered: за да кажете на играча колко струва продуктът. Учениците могат да използват блок Looks – thinking със слепения текст – стойността на променливата и тескт за пояснение, например:
think join Watermeloncests watermelon_price EUR ++ for 2 secs
 c) When clicked: тук трябва да направят малка рефлексия (да помислят). 1) В кой случай играчът може да закупи продукта и в кой не? 2) Какво се случва с бюджета, ако той купи храната? 3) Как да броим закупените продукти? 4) Какво се случва с храната на рафта?
 Играчът може да закупи продукта, ако има достатъчно пари. Така че учениците трябва да сравняват две променливи: budget and watermelon_price. Ако динята струва повече, отколкото има, той не може да я купи. Учениците могат да добавят малко текст, за да кажат на играча, че не може да купи този продукт.
 Ако играчът има 15 EUR и купи диня за 4 EUR, той вече има 15 - 4 = 11 EUR. Така че бюджетната стойност е сега: Стойността на предишнич бюджет budget value –
watermelon_price.
Студентите могат да добавят малко текст и тук.
say Great-choicel for (2) secs set budget to budget watermelon_price
 Преброяването на броя на закупените продукти ще се осъществи с промяна на променливата healthy_food va с 1. change healthy_food va (1)
4) Когато върху храната се щракне, тя се скрива.





hide
Едно възможно репение е:
when I am clicked
say You'don't have enough money. for (5) secs
say Great-choicel for (2) secs
set budget to [budgetwatermelon_price])
change healthy_lood by 1
[Стъпка 4]
За да има повече храна по рафтовете, учениците могат да
дублират спрайта на динята. Да приемем, че втората храна ще
бъде торта. Тогава кодът от [Стъпка 3] се нуждае от някои
промени. Учениците трябва да:
 Сменят костюм Да създадат нова променлива: <i>cake_price</i> Да присвоят някаква стойност на промвнливата cake_price Да променят в кода всеки блок, съдържащ променливата <i>watermelon_price</i> с <i>cake_price</i> Променят отговора при закупуване на тортата Да заменят <i>change healthy_food by 1</i> c <i>change unhealthy_food by 1</i>
E.g., За тортата кодът към when clicked e:
when I am dicked = If cake_price > budget say You'don't have enough money. for (5) secs else say Toormuch sugar! for (2) secs set budget = to budget = cake_price change unhealthy_food = by (1) hide
[Стъпка 5]





Когато играчът приключи покупката си, той щраква върху бутона
Finish. За да кажем на програмата, че играчът е щракнал върху
бутона (приключил с купуването на храна), разпространяваме
съобщение.
when I am clicked + broadcast finish +
[Стъпка 6]
В края се връщаме към спрайта на момичето.
Когато играчът завърши пазаруването си, искаме момичето да му
каже колко здравословни и нездравословни продукти е купил.
Когато играчът щракне върху бутона за финал, се изпраща
съобщение за финал. Когато момичето получи финала на
съобщението, тя казва, напр. "Избрахте Х здравословни продукти
и Y нездравословни продукти".
when I receive finish say join Yourchose healthy_food healthy-products and unhealthy_food for unhealthy-products. () 5 secs
[Стъпка 7]
По всяко време по време на играта, играчът може да провери
бюджета си, като постави мишката върху момичето. Например тя
може да каже / мисли нещо като:
when I am mouse-entered say join YOUTAVE budget EUR (* for 2 secs





[Фи	нален код]
Girl	
	<pre>en le clicked budgel = to [] bealthy_lood = to</pre>
	y join You'have budget EUR. () for (2) secs
Foo	d
	en clicked ow t cake_price to 3
	hen 1 am clicked - budget
	ay Yourdon't have enough money. for (5) secs ay Toormuch sugar! for (2) secs tet budget = to budget = cake_price thange unhealthy food = by (1)
	lide
Fini	sh Button





	when I am clicked = broadcast finish =		
	[Допълнителни задачи]		
	Учениците могат да добавят допълнителни задачи според техните желания или могат да следват задачите по-долу:		
	 Променете играта, за да можете да купувате всяка храна 3 пъти. 		
	 Дайте повече пари на играча в началото. В края момичето разказва също колко продукти сте закупили. Напр. "Купихте 2 пъти диня, 1х грозде, 2 пъти пържени картофи". 		
Инструменти и	 Цялата дейност в Snap!: 		
ресурси за учителя	https://snap.berkeley.edu/project?user=mateja&project=Buyi		
	ng%20food%20for%20a%20picnic		
	 Допълнителните задачи (възможни решения): 		
	https://snap.berkeley.edu/project?user=mateja&project=Buyi		
	ng%20food%20for%20a%20picnic%20%2B%20Add.%20Task		
	 Lajovic, S. (2011). Scratch. Nauči se programirati in postani računalniški maček. Ljubljana: Pasadena. Vorderman, C. (2017). Računalniško programiranje za otroke. 		
	Ljubljana: MK.		
Инструменти и	Инструкции за учениците		
ресурси за	(C4G16_InstructionsForStudent_BG.docx)		
учениците			





Учебен сценарий 17 – Операции

Учебен сценарий	Операции	
Предишен опит с програмирането	 Използване на променливи за преброяване на точки в игра и за избор на костюм на сцената и спрайта; Използване на случайно число за избор на костюм на сцената и на спрайта. Използване на цикъл за повторение Използване на условни блокове Използване на операции за сравнение Използване на блок за диалогов прозорец (попитайте и изчакайте) 	
Резултати от	Общи резултати от обучението:	
обучението	 Променливи Условни блокове Цикъл Сензорни блокове Синхронизиране на действията чрез съобщения 	
	Специфични резултати от обучението, ориентирани към	
	алгоритмично мислене:	
	 Ученикът използва променливи за преброяване на точки и за костюми на сцената и на запазването на спрайт. Ученикът използва променливи за преброяване на точки Ученикът инициализира променливи за преброяване на точки Ученикът използва условни и логически операции Ученикът използва съобщения за промяна на спрайта и изчисляване на крайния резултат. 	
Цел, задачи и	Кратко описание:	
кратко описание на	Нека с игра да проверим дали играчът е усвоил аритметичните	
дейностите	операции в Snap!. Правилата са следните: Десет пъти по-случаен начин се избира аритметична операция с първи операнд 6 и по случаен начин се избира втори операнд – число от 1 до 3. Играчът трябва да въведе верен отговор. Броят се верните и грешните отговори. В края на играта се съобщава какъв е верният резултат. Задача: Студентите трябва да определят декора на сцената и костюмите на спрайта; да планират необходимите променливи; да определят какви блокове са им необходими. На финала трябва да	





	създадат кодовете към сцената и спрайта.	
	Допълнителните задачи могат да са: в зависимост от получения	
	резултат, спрайтът да казва "Справи се чудесно!", "Все още не	
	познаваш добре аритметичните операции в Snap!" и т. н.	
	Цел: Учениците ще подобрят своите вече придобити знания за	
	променливи, случайни числа, цикли, съобщения.	
Продължителност	45 минути	
Стратегия и методи	Активно учене (дискусии, експеримент с вече подготвена игра),	
на обучение	обучение, основано на програмиране на игра, решаване на проблеми,	
Форми на	Самостоятелна работа / Работа по двоики/ Презентация пред	
обучение	класа/групата	
Ход на урока	(Мотивация-Въведение, Прилагане, Осмисляне и Оценка)	
	Учителят поставя проблема за необходимостта от игра, която да се	
	установи дали са усвоени добре аритметичните операции в Snap! и	
	демонстрира проекта.	
	https://snap.berkeley.edu/project?user=ddureva&project=operations3.	
	operations3 by ddateve	
	Correct Writing Writing	
	 Обсъжда се как да се формулира условието на задачата. Прави се формулировка на задачата. Десет пъти по-случаен начин се избира аритметична операция 	
	с първи операнд 6 и по случаен начин се избира втори операнд –	
	число от 1 до 3. Играчът трябва да въведе верен отговор.	
	Броят се верните и грешните отговори. В края на играта се	
	съобщава какъв е верният резултат.	
	2. Коментират се променливите и начина на тяхното дефиниране,	





инициализиране и промяна на стойността им.

- 3. Припомнят се командите за random number, аритметичните и логическите операции, broadcast event
- 4. Обсъжда се дали основният код да е към сцената или към спрайта. В примера основният код е към сцената, а в кода на спрайта има скриптове за смяна на костюм и изчисляване на крайния резултат.



В кода на сцената се прави инициализация на променливите за верни и грешни отговори.

За избор на операция се използват командите:

set operation - to pick random (1) to (4) switch to costume operation

Избора на костюм за спрайта се осъществява с broadcast to Number Sprite. избраният номер на костюма се съхранява в променливата CostumeNumber, която е дефинирана за всички обекти в проекта и затова може да се използва в кода на сцената.

Код към сцена	Декори на сцена	





Ö operations3	Ø operations3
	exes Stage
oseno Stage	Suph Exhauseds Sands
Cosinte Backersunde Sounds	per per
scripts backgrounds sounds	e o
when 🔁 clicked	Emaily
set wrong to 0	a file an year computer by dropping it have
repeat 10	0
set operation - to pick random (1) to (4)	NR
switch to costume operation	
broadcast Change costume number	COMIN
ask inputionswer! and wait	· · · · · · · · · · · · · · · · · · ·
if costume name of Stage = =	Canco
if (6 - CostumeNumber) = answer	**
change correct by	6787
change wrong w by 1	
If costume name of Stage = =	
if 6 + CostumeNumber = answer	
else	
change wrong wr	
<pre>if costume name of Stage = = ` if @ x CostumeNumber = answer change correct by 1 else change wrong v by 1 if costume name of Stage = ? if @ CostumeNumber = answer change correct by 1 else change wrong v by 1 else change wrong v by 1 else change wrong v by 1</pre>	
след като са изорани по случаен начин д	стора па сцепата и костюма
на спраита се поставя въпрос към играч	ча за въвеждане на верния
отговор от операцията с командата.	
ask Inputanswer! and wait	
Въведеният отговор се сравнява с резулта	та от избраните операции.
Използва се командата	





if (условие)

else

Ако е избрана операция "-", то се прави проверка дали 6 минус "номера на костюма на спрайта" съвпада с отговора. Ако съвпада се увеличава променливата correct, иначе се увеличава променливата за броене на грешни отговори wrong.



За останалите команди скриптът е подобен, разликата е в избраната операция.

За да се спести подреждането на кода за останалите операции може

да се покаже как се копира част от кода и се променя аритметична



Копиране на код:

- 1. Щракни с десния бутон на мишката върху кода
- 2. От контекстното меню избери Choose duplicate (дублирай)

relabel	and pick it up	umber	= answer
delete script pic	rect by (1	
ringify	1.000		

3. Постави с мишката дублирания скрипт на съответното място.

Учителят може да постави задача учениците сами да открият как да копират част от кода.

Промяна на операцията.

1. Щракни с десния бутон на мишката върху дадена операция. Ще с епояви контекстно меню.





	 исстраната об Stage = : иличение степенуване (^) и деление по модул (mod) Учениците работят в екипи като създават собствени костюми за сцената и спрайта. При ограничения във времето може да се използва "half backed" project, който съдържа готовите сцена и спрайт.
Инструменти и	Цялата дейност е в Snap!:
ресурси за учителя	https://snap.berkeley.edu/project?user=ddureva&project=operations3
	Цялата дейност е в Scratch:
	 Дурева Д., М. Касева, Г. Тупаров, Компютърно моделиране, 4. клас, Просвета, 2018, София (Dureva, D., М. Kaseva, G. Tuparov, Kompyutarno modelirane, 4. klas, Prosveta, 2019, Sofia)
Ресурси/материали	https://snap.berkeley.edu/project?user=ddureva&project=operations half
за ученика	Инструкции за учениците
	(C4G17_InstructionsForStudent_BG.docx)





Учебен сценарий	Рециклиране	
Предишен опит с	• Показване и скриване на спрайт	
програмирането	• Използване на променливи за преброяване на точки	
nporpannpanero	• Използване на цикъл завинаги	
	 Използване на условни блокове 	
	• Използване на операции за сравнение	
	• Използване на чувствителност на цветовете	
Резултати от	Общи резултати от обучението:	
обучението	• Променливи	
	• Условни блокове	
	• Цикъл	
	• Придвижване до точка в посока	
	• Сензорни блокове	
	• Рефакторинг на код	
	Специфични резултати от обучението, ориентирани към	
	алгоритмично мислене:	
	 Ученикът използва Изчакай до (Wait until) и логически 	
	операции за приключване на играта	
	• Ученикът използва Изчакай до (Wait until) и блок за	
	промяна на сцената	
	 Ученикът използва променливи за преброяване на точки 	
	• Ученикът използва условни и логически операции	
	 Ученикът сравнява кодовете на подобни спрайтове. 	
	 Ученикът прави рефакторинг на кода 	
	• Ученикът използва позициониране на спрайтовете (в	
	допълнителна задача използваите произволно	
	позициониране)	
цел, задачи и	пратко описание:	
кратко описание на	Някой е разхвърлял боклук пред училището. Играчът е помолен	
дейностите	да помогне за разделно събиране на боклука като го сортира за	
	рециклиране на хартия и стъкло. Когато се поставя боклук в	
	правилния контейнер, то боклукът се скрива. Ако боклукът се	
	поставя в неправилен контейнер се появява съответното	
	съобщение – "Това не е контейнер за хартия." или "Това не е	
	контейнер за стъкло" и боклукът се връща на първоначалната си	
	позиция. Играта приключва когато всичкия боклук е поставен в	





	правилните контейнери.	
	Задача: Учениците трябва да изследват кодовете на сцената и	
	спрайтовете, да сравняват кодовете от типа хартия за отпадъци и	
	отпадъци от стъкло, да добавят нови спрайтове и код и да	
	променят кода в сцената по отношение на новите добавени	
	спрайтове. Допълнителни задачи могат да бъдат към:	
	• промяна на положението на отпадъчните спрайтове с	
	произволен избор на координати на спрайтовете;	
	• намалете броя на етапите и извлечете робота като отделен	
	спрайт. (Роботът е част от фона на сцената).	
	Цел: Учениците ще подобрят своите вече придобити знания и	
	ще разширят сценария на играта с нови обекти, код и променящ	
	се код по отношение на нови спрайтове. Те ще бъдат обучени да	
	правят рефакторинг на код.	
Продължителност	45 минути	
Стратегия и методи	Активно обучение (дискусии, експеримент с вече подготвена	
на обучение	игра), обучение, основано на игрален дизайн, решаване на	
	проблеми.	
Форми на обучение	Самостоятелна работа / Работа по двойки/ Презентация пред	
	класа/групата	
Резюме на	(Мотивация-Въведение, Прилагане, Осмисляне и Оценка)	
обучението	Учителят поставя проблема за разделно събиране на боклук и	
	коментира какви са цветовете на кофите за различните видове	
	боклук – син за хартия, зелен за пластмаса.	
	1. Поставя за задача учениците да изиграят играта и опишат с	
	думи: Колко сцени наблюдават и колко спрайта (герои)? Как	
	започва играта? Кой спрайт пита за името на играча? Колко	
	променливи се използват и как са наименовани? Какво се	
	случва когато хартия се постави в контейнер за стъкло и какво	
	когато се постави в контейнер за хартия?	







2. Актуализация на изучавани команди

Припомнят се командите за осъществяване на диалог с потребителя.

Прави се коментар за смяна на сцените – Сцена 1 с Робота, Сцена 2 с училището и разхвърляния боклук и Сцена 3 с Робота и надписа Bravo!. Обсъждат се възможните команди за смяна на сцена.



Дискутира се, че проверката за правилното поставяне на боклука в даден контейнер трябва да се осъществи с условен блок (conditional block) и блокове с условия за допир от групата Sensing. Прави се словесно описание: Ако даден боклук-хартия допира кофата за хартия, то боклука се скрива (поставен е в правилната кофа) и точките за прибрана хартия се увеличават с 1.











me paper4.	paper1
Scripts Costumes Sounds	Sorigin Costumes Sounds
when Clickod bide	when Clicked hide
well until costume name of State = school show	show
go to x: (103) v: (86) terever	go 10 x: 729 v: -141
if Couching class on 2 say This is not paper bin for (2) secs	say This is not paper bin for 2 secs
co to x: <103 v: <85	op to x: (79 v: (14)
hide change operative (hide change coord by D
stop the scott "	stop min acrost
No. 1944	diass4
W daggable	er / rreggable
when a circles	when a clicked
hide well costume name - of Stage	hide well well continue name of State - School
show go to x: -62 v: -133	show go to x: 6 y: 6162
Increase: If Touching poperton = 7	i touching operain = ?
say Thistanotolasabin for (2) secs go to x: (82) y: (133)	say This is not class bind for (2) secs on to x: (6) y: (102)
if (Inuching cluss bin w P)	i Couching gass ton w 7
change gass the 1	chanue canne by
4. Поставяне на задача за	допълване с два нови спрайта –
ооклук-хартия и ооклук- промяна в кодовете на сі	стъкло, задаване на код към тях и цената и контейнерите за боклук.
Коментира се как да създад	цат двата нови спрайта. Варианти –
дублиране на съществуващи	и редактиране в Snap!, създаване на
нови в графичен редак	гор или търсене на свободно
разпространявани изображен	ия в интернет и импортирането им в
играта.	
Необходимо е да се направи	коментар и за промените в кода на
сцената по отношение на завт	ьршването на играта.
Да се коментира възможн	о ли е задаването на началните
стойности на променливит	е да бъде не в кода на двата
контейнера, а в кода на с	цената и съответно да се направи





корекция.

По преценка на учителя може да се усложни условието на задачата:

 при всяко стартиране на играта боклукът да се разпръсва на произволно място. Тук е добре да се обърне внимание, че трябва да се ограничат координатите в които би могъл да се разпръсне боклука, така че да бъде на реалистично място. Например ограничен от координатите на червения правоъгълник.



 Да се въведе нов спрайт Робот и да се намали броят на декорите на сцената. Да се напише съответния код към Робота, така че той да води диалога с играча, а не спрайта син контейнер.

	син контеинер.
Инструменти и	Цялата дейност е в Snap!:
ресурси за учителя	https://snap.berkeley.edu/project?user=ddureva&project=recycling
	Цялата дейност е в Scratch:
	 Дурева Д., М. Касева, Г. Тупаров, Компютърно моделиране, 4. клас, Просвета, 2018, София (Dureva, D., M. Kaseva, G. Tuparov, Kompyutarno modelirane, 4. klas, Prosveta, 2019, Sofia)
Ресурси/материали	https://snap.berkeley.edu/project?user=ddureva&project=recycling
за учениците	Инструкции за учениците
	(C4G18_InstructionsForStudent_BG.docx)





Учебен сценарий 19.1 - Свири на пиано

Учебен сценарий	Свири на пиано	
Предишен опит в	Използване на променливи за преброяване на точки;	
програмирането	Използване на събитие Когато съм под натиск	
	Използване на цикъл за повторение (repeat loop)	
	Използване на условни блокове	
	Използване на съобщения за промяна на костюмите на сцената и	
	за управление на дейностите на спрайта	
Резултати от	Общи резултати от обучението:	
обучението	• Променливи	
	• Условни блокове	
	• Цикъл	
	• Съобщения;	
	• Звуци;	
	• Програмиране на музика;	
	Специфични резултати от обучението, ориентирани към	
	алгоритмично мислене:	
	• Ученикът използва променливи за преброяване на точки.	
	• Ученикът инициализира променливи за преброяване на точки	
	• Ученикът използва условни блокове, за да оцени постигнатите	
	точки	
	• Ученикът използва съобщения събитие за смяна на костюмите	
	на сцената и за действията на спрайтове.	
	• Ученикът използва блокове от Група Звук, за да композира	
	мелодии;	
	• Ученикът идентифицира необходимостта от цикъл за	
	повторение, за да намали броя на блоковете в скриптовете.	
	• Ученикът разширява функционалността на играта.	
Цели, Задачи и	Кратко описание:	
кратко описание на	Да се пренесем в чудния свят на Кралица Мери. Тя кани играча в	





дейностите	двореца си да послуша музика. В балната зала нейният приятел
	динозавърчето Дино свири на пиано. В играта Дино свири
	няколко музикални тона, а играчът трябва да познае кой тон
	свири Дино. Ако познае получава една точка за правилен отговор,
	ако не познае една точка за неправилен отговор. След
	разпознаването на тоновете се поставя по-сложна задача: Дино
	свири на пианото мелодия, а играчът трябва да познае мелодията
	от коя песен е. За правилно позната мелодия получава 5 точки.
	Задача: Учениците използват "half backed"(полуготов) файл с
	декорите на сцените и спрайтовете. Трябва да планират
	необходимите променливи, да определят какви блокове са им
	необходими. Да се запознаят с блоковете от групата Sound и
	начина на изсвирване на ноти. Да създадат скриптове, с които да
	се изсвирят няколко мелодии.
	Цел: Учениците ще научат за кодиране и възпроизвеждане на
	мелодии и ще подобрят своите вече придобити знания за
	променливи, цикли, условни блокове и операции, излъчвани
	събития и други събития.
Продължителност	90 минути
Стратегия и методи	Активно обучение (дискусии, експеримент с вече подготвена
на обучение	игра), обучение, основано на игрален дизайн, решаване на
	проблеми.
Форми на обучение	Самостоятелна работа / Работа по двойки/ Презентация пред
	класа/групата





Ход на урока	(Мотивация-Въведение, Прилагане, Осмисляне и Оценка)
	1. Учителят поставя задача за създаване на играта. Обсъжда се с
	какви средства може да се реши задачата. Стига се до извода, че
	към момента не познават възможностите за писане на код за
	изпълнение на музика.
	2. Учителят демонстрира част от играта с композиране на
	мелодия.
	https://snap.berkeley.edu/project?user=ddureva&project=Play a Pia
	<u>no 1</u>
	Play_a_Piano_1 by iddresse
	Which is the note C or F?
	1. Показва кода и обяснява използването на командите от
	групата Sound.
	В Snap! може да се използват звукове от вградената
	библиотека, файлове от компютъра или да се свирят
	музикални тонове на различни инструменти.
	За избор на инструмент се използва командата:
	(1) sine (2) square (3) sawtooth (4) triangle , Забележка. В Scratch инструментите са
	много повече.
	Учениците тестват звука на различните инструменти.
	2. Обяснява начина на задаване на нотите:
	Използва се командата: play note 60 for 0.5 beats. В нея
	първото число задава нотата, а второто число описва за колко




такта се изпълнява нотата.

Когато се щракне върху стрелката до първото число се появява клавиатура на пиано и от нея може да се избере нота. Пианото обхваща две октави



С	C#	D	Eb	Е	F	F#	G	G	А	Bb	В	С
								#				
60	61	62	63	64	65	66	67	68	69	70	71	72

Продължителността на всяка нота се задава с числа 1 – цяла нота, 0.5 - половинка, 0.25 - четвъртинка. (За ученици, които не са изучавали реални числа може да се дадат чрез операцията деление – ½, ¼, 1/8 и т.н)

play note 59 for 1 / 4 beats play note 60 for 1 / 2 beats

По преценка на учителя, учениците може да експериментират с командите и да установят сами зависимостите.

3. Прави се коментар на скрипта за мелодията Jingle Bells като се използва и нотния запис на мелодията.







	се повтарят. Коментира се коя команда трябва да се използва.
	(repeat loop)
	4. Учениците се разделят на екипи, които трябва да създадат
	играта поставена в началото на урока. Всеки екип обсъжда
	сценария на играта и описва плана на играта в листа за
	описание (Приложени SNAP_Program_Design_and_Planning
	Worksheet.docx) В описанието може да се добавят таблици
	за детайлно описание на действията в сцените и на
	спрайтовете. Може да се добави условие – динозавърът да
	танцува, докато свири. (В предварително подготвения файл
	динозавърът има няколко костюма)
	5. Учителят може да покаже част от сценарии от файла
	https://snap.berkeley.edu/project?user=ddureva&project=PlayA
	<u>Piano</u>
Инструменти и	Цялата дейност е в Snap!:
ресурси за учителя	https://snap.berkeley.edu/project?user=ddureva&project=Play a Pia
	<u>no 1</u>
	https://snap.berkeley.edu/project?user=ddureva&project=PlayAPiano
Ресурси/материали	https://snap.berkeley.edu/project?user=ddureva&project=Play a Pia
за ученика	no Half backed
	Инструкции за учениците
	(C4G19.1_InstructionsForStudent_BG.docx)





Учебен сценарий	Свири на пиано				
Име					
Предишен опит в	 Използване на цикъл repeat 				
програмирането	• Използване на променливи				
	 Използване на условни блокове 				
Резултати от	Общи цели на обучението:				
обучението	• Условия				
	• Цикли				
	специфични цели на обучението, ориентирани към алгоритмичното мислене:				
	 Ученикът използва цикъл с повторение за възпроизвеждане на музика 				
	 Ученикът използва код, за да накара спрайтовете да реагират на въвеждането 				
	 Ученикът добавя звущи към спрайт 				
	 Ученикът използва код, за да промени костюма на спрайт 				
Пел залации					
кратко описание на	прятко описание. Ученикы трябва да изсвири песен на пиано по				
дейностите	дадени ноти.				
	Задача: Трябва да програмирате клавишите за пиано - всеки клавиш				
	трябва да свири определен тон. На сцената трябва да се покажат				
	два различни бутона, единият за показване на нотите, а другият за				
	възпроизвеждане на мелодията.				
	Цел: Учениците ще се научат как да свирят музика и да сменят				
	костюма, като кликнат върху спрайт.				
Продължителнст	45 минути				
Стратегия и методи	Активно учене; обучение, основано на програмиране на игра;				
на обучение	решаване на проблеми				
Форми на обучение	Самостоятелна работа / Работа по двоики				
Ход на урока	(Мотивация-Въведение, Прилагане, Осмисляне и Оценка)				
	В началото на сцената се показва пиано. До пианото ще има два				
	бутона. Щракването върху първия бутон трябва да показва нотите и				

Co-funded by the Erasmus+ Programme of the European Union





думите на песента, а щракването върху втория бутон трябва да възпроизвежда мелодията, която трябва да се изсвири. Освен това до пианото ще бъде бутонът "Х", който ще рестартира проекта. [Стъпка 1]

Отворете програмата *Play a piano.* Програмата съдържа всички фонове и спрайтове, необходими за тази задача.

Даден е фонът, а също и спрайтът за клавиш С и един черен клавиш. Учениците трябва да дублират ключа С, да го преместят в правилната позиция и да го преименуват. Клавишите трябва да са в следния ред: С, D, E, F, G, A, B. Клавиатурата трябва да изглежда както на следващата картинка и да възпроизвежда тонове, написани под клавишите:



Дублирайте спрайт "black_key" 4 пъти, за да получите 5 черни клавиша, и ги именувайте от черен ключ 2 до черен ключ 5. Поставете нови черни клавиши между клавишите D и E, F и G, G и A и A и B.

Ако черният клавиш е скрит зад белите клавиши, използвайте следния код

go to front a layer

Направете същото за клавиша В и ги поставете в края на клавиатурата.

На първо място, трябва да сте сигурни, че бутонът "draggable" е отменен за всички спрайтове, за да не могат да бъдат премествани по време на игра.







[Стъпка 2]

След това активирайте възпроизвеждането на тонове чрез натискане на спрайтове. За бутона "С" добавете блок "When I am clicked" и го оставете да излъчва съобщението "с", както следва:



Въпреки това, за да издадете звук, когато натиснете клавиш, добавете блок "When I receive с" и добавете бележка 60 за 0.5 удара.



За да подчертаете кой клавиш е натиснат, трябва временно да промените костюма на този спрайт. За да направите това, в блока "When I am clicked" сменете костюма на с1 за 0,2 секунди, след което се върнете към костюм с.



[Стъпка 3]

Повторете това кодиране за другите бели клавиши.

Задайте за другите клавиши следните бележки: D – 62, E – 64, F – 65,

G – 67, A – 69, B – 71

[Стъпка 4]

Сега, за да свирите на пианото с клавиатурата, добавете блок "When c key pressed", за да натиснете клавиша с Sprite, и копирайте останалата част от кода от блока "When I am clicked".





when a key pressed broadcast a switch to costume a1 wait 02 secs switch to costume a

Забележете, ако клавиша "с" на клавиатурата е задържан, звукът ще се повтаря, докато клавишът е натиснат. Това се случва, защото съобщението "с" се излъчва многократно. За да спрете излъчването на съобщение, в края на кода добавете блок "изчакайте до""wait until" от палетата за контрол.



За да завършите излъчването на съобщение, използвайте оператора "not" и добавете блок "key c pressed".



Направете същото за останалите бели клавиши.

[Стъпка 5]

Време е да програмирате петолинието, ключът сол и нотите, които ще се изпъляват:



За да покажете нотите, активирайте излъчването на съобщението "chords", когато натиснете бутона.



Импортирайте нов костюм "chords" за тази сцена.











Co-funded by the Erasmus+ Programme of the European Union

and the second se
when I am cloud a
repeat (2)
play note 60 for 0.5 beats
repeat 🙃
play note GTP for GTP beats
Contracting Contracting Contracting
mpeut 🕘
play note (25) for (25) beats
play note CTP for CCP beats
party notes and the day hereits
reprint 🔁
play note GID for CED beats
play note 62 for 616 beats
play note (0) for (0.5) beats
wait (0.1) sees
regieut 🕢
play note (67.) for (0.5) beats
play note 610 for 600 beats
play note (05) for (05) heats
play note (153) for (153) heats
play note (557) for (558) heats
play note (AT) for (AT) brats
when we the first for first hands
wait OI sers
repeat 🕢
play note 600 for 605 beats
play motine of the local boats
repeat 🕣
play note (20) for (25) beats
play note 67 for 0.5 beats
wait (0.1) sees
repeat 🕘
play note 65 for 0.5 beats
repeat 📵
play note C49 for C59 beats
repeat 😧
play note (22) for (05) beats
play note (00) for (0.5) beats





[Стъпка 7]
След това програмирайте спрайта "Х" по начин, който ще нулира
проекта (без забележките), както следва:
 задайте размера му на 50%. активирайте излъчването на "blank" съобщение при натискане на бутона.
set size to 50 % broadcast blank
В края добавете към сцената блок "When I receive", за да
промените костюма към "blank", след като получите съобщението
when I receive chords when I receive blank and a switch to costume chords and switch to costume blank and a switch to costume
[Допълнителни задачи]
Учениците могат да добавят допълнителни задачи според техните
предпочитания или могат да следват задачите по-долу:
 Дублирайте спрайт нотата (и променете позицията й на фона) и напишете програма за друга песен. Добавете фон с акорди за новата песен.
[Финален код]
А клавиш
when I am clicked when a week pressed broadcast a witch to costume a witch to cost a
Клавиш цигулка
when I am clicked - broadcast chords -





Co-funded by the Erasmus+ Programme of the European Union

Нота

when I am did	kast 🐨
repeat (2)	
play note 60	for (15) beats
repeut 💿	
play note 67	for 01 beats
repeut 2	
play note 🚱	• for (0.5) beats
- Andrews	Con COE Instan
	And a state of the second state
repeat @	-
play note (85	• for OS beats
Der stopenteren an	A COLORADOR
C. Maning	And an Antonia
repeat (2)	
play note 62	• for (0.6) beats
play note 60 .	for 0.5 beats
wait 0.1 see	
repieut 🕢	-
play note 67	• for 0.5 beats
play note 67	• for 0.5 beats
play note 65	for 0.5 beats
and the state of the	a for (15) boots
niav note A	for file beats
play note 63	for (15) beats
wait 0.1 see	CS
1	
repeat 2	No. of Concession, Name
play note 60	for (15) beats
repeat 🕥	
play note 61	for 0.5 beats
repeat 🕢	
play note 69	• for (15) beats
Anna and an	
party note d/	No. 0.5 Beats
renneral C	
play pote AS	• for (03) beats
a start the start of the	and the second s
repeat (2)	-
play note 64	for 0.5 beats
repeat 😩	
play note 🚳	• for (05) beats
and the second se	





	x
	when clicked when I am clicked set size to 50 % broadcast blank
	Сцената
	when I receive blank switch to costume blank
Инструменти и	Цялата дейност в Snap!:
ресурси за учителя	https://snap.berkeley.edu/project?user=ifrankovic&project=Play%20a%
	20Piano
Ресурси/материали за	Полуготов файл в Snap!:
ученика	https://snap.berkeley.edu/project?user=ifrankovic&project=Play%20Pia
	<u>no</u> (27.1.2020)
	Изображения:
	 Sprite изображения:
	 a.png, a1.png
	 b.png, b1.png vialing language
	 violin_key.png Backgrounds: notes nng
	 Инструкции за учениците (С4С10.2. InstructionsEarStudent_PC docx)





Учебен сценарий 20 - Тест

Учебен сценарий	Тест
Име	
Предишен опит в	• Показване и скриване на спрайтове
програмирането	 Употреба на променливи за преброяване на събраните точки
	 Използване на цикъл forever
	 Използване на условни блокове
	• Използване на оператори за сравнение
	 Използване на сензорни блокове
	• Промяна на сцена
Резултати от	Общи цели на обучението:
обучението	• Променливи
	• Условия
	• Цикъл
	• Сензорни блокове
	Специфични цели на обучението, ориентирани към
	алгоритмичното мислене:
	• Учениците използват условия за да преценят отговора -
	правилно или грешно
	 Учениците използват блокове за смяна на костюма на сцената
	 Учениците използват променливи за броене на точки
	• Учениците използват логически операции
	 Учениците използват външен графичен редактор за подготовка на сложни фонове на сцените.
Цел, задачи и	Кратко описание: Помогнете на вашия учител да тества вашите
кратко описание на	знания за Snap!, като създадете игра, базирана на
дейностите	Quest(Въпросник), за да тествате командите в Snap!
	Задача: Трябва да проучите примерна игра, затова изберете от
	полуготовите игри, намерете или създайте свой собствен спрайт
	(Sprite), който да задава въпроси, изберете от полуготовите игри
	или проектирайте фона на началния екран и сценичните фонове с
	подходящи въпроси, променете и разщирете кодовете в теста по
	отношение на въпросите.
	Цел: Учениците ще подобрят знанията си и ще разширят сценария





	на играта с нов фон, код и променлив код по отношение на				
	новите сцени.				
Продължителнст	90 минути				
Стратегия и методи	Активно учене (дискусии, работа с полуготови игри); обучение,				
на обучение	основано на програмиране на игра; решаване на проблеми				
Форми на обучение	Самостоятелна работа / Работа по двоики / Директен контакт с цял				
	клас				
Ход на урока	(Мотивация-Въведение, Прилагане, Осмисляне и Оценка)				
	1. Трябва да създадете игра-тест, за да тествате знанията си по				
	програмиране.				
	2. Възлага се на учениците да играят играта и да опишат с думи:				
	Колко сценични декорации наблюдават и колко спрайтове (герои)?				
	Как започва играта? Колко променливи се използват, как се				
	именуват, за какво се използват? Какво се случва, когато отговорът				
	е верен/грешен? Как са представени въпросите в теста?/				
	индивидуална работа или работа по двойки по преценка на				
	учителя/				
	3. Трябва да използвате алгоритъма за задаване и отговор на				
	въпроси:				
	1. преминаване към сценичен костюм (съдържа въпрос);				
	2. задаване на костюм на Аби за задаване на въпрос;				
	3. Аби казва - Отговорете с Да или Не;				
	4. Играчът въвежда отговор - Да или Не;				
	5. Ако отговорът е верен, Аби казва "Правилен отговор" и				
	броят на верните отговори се увеличава; В противен случай				
	Аби казва "Грешен отговор" и броят на грешните отговори се				
	увеличава.				
	4. Какво се случва след като отговорите на всички въпроси?				
	1. Смяна на костюм / фон на сцената;				
	2. Аби посочва броя на правилните и грешни отговори и дава				
	оценка.				











Abby
when 1 receive State Sav Westerl for @ secs rm switch to costume 100 0 ftbor switch to costume abor switch to costume abor
Коментират се ситуациите, когато верният отговор е "да" и когато
верният отговор е "не".
Кодът за оценка е описан подробно, като е използвана
променливата Total.
switch to costume abby-a rum switch to costume room2) of Stage) say Thenumberof correct answers is: for 2 secs say Correct for 2 secs say Thenumberof wrong answers is: for 2 secs say Wrong for 2 secs set Total to Correct - Wrong say ioin Total score is Total () for 2 secs if Total > 2 switch to costume abby-c say Yourerdoing well eise switch to costume abby-b say Readyourlessons againt
Обсъжда се начинът на проектиране на сцената за отделни
въпроси.
Тъй като в Snap! не е възможно да се пише текст в костюми и
декори, е необходимо да се използва външен графичен редактор.





	Друга възможност е да използвате MS Powerpoint, за да създадете				
	въпроса и да експортирате съответното текстово поле в графичен				
	формат.				
	Добавяне на костюм в Snap! Може би преработен.				
	1. Разделяне на групата на отбори от 2 или 3 ученика.				
	2. Публикуване на темата за тестовите въпроси. Например -				
	Използване на променливи; Цикли; Движение, Сензори,				
	аритметични и логически операции.				
	3. Проектиране на сцени с въпроси по тема от съответния екип.				
	Ако е необходимо, учителят съветва учениците по съдържанието				
	на въпросите. Въпросите се обсъждат и всеки екип създава сцена				
	за поне два въпроса.				
	4. Създаване на кода. На студентите се предоставя полуготов файл				
	с костюми на сцената и спрайтове. Те също могат да създадат свой				
	собствен файл, ако желаят. Работата се извършва по аналогия с				
	теста на модела.				
Инструменти и	Цялата дейност в Snap!:				
ресурси за учителя	https://snap.berkeley.edu/project?user=ddureva&project=test2				
	Цялата дейност в Scratch:				
	• Дурева Д., М. Касева, Г. Тупаров, Компютърно моделиране,				
	Ч. клас, просвета, 2018, София (Dureva, D., M. Kaseva, G. Tuparov, Kompyutarno modelirane, 4. klas, Prosveta, 2019,				
	Sofia)				
Ресурси/материали	 Полуготов файл в Snap!: 				
за ученика	https://snap.berkeley.edu/snap/snap.html#present:Username=spelac&				
	ProjectName=C4G 20 test en tmp				
	• Инструкции за учениците				
	(C4G20_InstructionsForStudent_BG.docx)				





Учебен сценарий 21 - Опростена игра РАСМАN

Учебен сценарий	Опростена игра PACMAN
Име	
Предишен опит в програмирането Резултати от обучението	 използване на условни блокове, кодиране на множество обекти, сензорни блокове, цикли (forever, repeat until), синхронизиране на действията чрез съобщения, използване на произволни числа Общи цели на обучението: Движение на обект базирано на събития, Засичане (разпознаване) на един цвят, Употреба на булеви стойности в логически изрази, Определяне, диференциране, динамично проверяване и реагиране на две различни състояния на играта. Специфични цели на обучението, ориентирани към алгоритмичното мислене: Ученикът осъществява движение на обект с клавишите със стрелки, използвай сензорен цветен блок, за да получи булева стойност за четене на едноцветни сензори, Ученикът осъзнава, че състояниято на обекта може да бъде изразено с цвета на обекта до който се докосва, Ученикът прави разлика между две различни състояния и знае как да ги изрази с логически изрази, Ученикът осъзнава, че позицията на обекта се променя динамично и използва цикъл forever за многократна проверка на текущото състояние,
	 Ученикът използва if-else, за да даде различни отговори въз основа на текущата позиция на обекта.
Цел, задачи и	Кратко описание: В този курс ще програмирате игра, в която
кратко описание на	главният герой трябва да вземе произволно разположени обекти,
дейностите	в случая звезди и ще бъде преследван от призрак.
	задача: Движението на главния герой трябва да бъде
	програмирано, така че тои да се движи в лабиринт. След това
	тряова да програмирате обектът звезда, които ще се клонира при
	стартиране на играта и ще се появява на произволно ново място





	всеки път, когато героят го вземе. Трябва да съхранявате броят на
	събраните звезди и да завършите играта, когато играчът събере 20
	звезди. За да направите играта по-интересна, можете да
	програмирате зъл призрак, който на случаен принцип ще се
	движи в лабиринта. Ако играчът докосне призрак, играта
	свършва.
	Цел: Учениците ще приложат своите знания за движение в
	лабиринт с помощта на сензорен цветен блок. Те ще научат
	концепцията за клониране на обект с ограничения за позицията
	и как да създадат много прост персонаж, който не е играч, с
	опция за произволно движение.
Продължителнст	90 минути
Стратегия и методи	Активно учене; обучение, основано на програмиране на игра;
на обучение	решаване на проблеми
Форми на обучение	Директно преподаване
	Самостоятелна работа / Работа по двоики/Работа в групи
Ход на урока	(Мотивация-Въведение, Прилагане, Осмисляне и Оценка)
	Играчът трябва да събират произволно разположени звезди,
	докато е преследван от червен призрак.
	Ако той и призракът се сблъскат, играта свършва, ако пък успее да
	съберете 20-те звезди, преди това играчът печели.
	[Стъпка 1]
	Инструктираме учениците да проектират лабиринт, в който
	зоната, където играчът има право да се движи, е от един цвят
	(например син) и стени, които спират движението на играча,
	оцветени в някакъв друг цвят (например черен). За да спестим
	време, можем предварително да подготвим фоновата картина на
	лабиринта.







[Стъпка 2]

Учениците трябва да нарисуват растап и червения призрак. За звезда можем просто да нарисуваме кръг в Snap!:



[Стъпка 3]

За да накарате Pacman да се движи, можете да използвате различни варианти. Един от тях е използването на система за събития за откриване на това, кой клавиш е натиснат, наляво, надясно, нагоре или надолу. След всяко от тези събития, трябва да тествате дали той докосва цвета на зоната, по която му е позволено да се движи. В този случай, най-напред той се обръща в тази посока и прави ход. Но ако докосне цвета на стените, трябва да се придвижи назад, защото в противен случай ще се забие в стената.







[Стъпка 4]

Следващата задача е да програмирате звездите. Звездите ще са еднакви и ще има много от тях. В този случай е по-добре, вместо да се направят множество идентични обекти (в нашия случай 20), да се направи един обект и след това да се клонира. В началото на играта първият клонинг ще се появи на случаен принцип в лабиринта, след това, когато го съберете, той ще изчезне и ще бъде създаден нов на различно произволно място. За да създадете първия клонинг в началото на играта, поставете този код на сцената:



А звездата трябва да бъде скрита when the green flag is clicked в началото на играта.

За да намерите подходящо произволно място за звездата, трябва да спазвате определени ограничения. Ако звездата е поставена върху стена, няма да можете да я достигнете, което означава, че не трябва да я поставяте там.





Стратегията за това е следната:

- Трябва да намерите случайна (х, у) позиция на клонинга (звездата). И двете координати, и х и у са в един и същи интервал [-140, 140]. Така че избирате произволно число от този интервал и за двете.
- След това проверявате дали този клонинг докосва цвета на стената. Ако е така, то местоположението му не е правилно.
- Ако местоположението е правилно, клонингът трябва да се покаже и да проверите чрез цикъл forever дали не се получава сблъсък с играча.
- 4. Ако местоположението не е правилно, създаваме нов клонинг и изтриваме този.
- За да преброите събраните клонинги, трябва да използвате общ брояч на звезди, които трябва да бъде дефиниран извън клонингите, например на играча.

Това може да стане чрез излъчване на съобщение, че сблъсъкът е настъпил. След това можете да го изтриете.



когато се блъсне в стена. За да направите движението му





случайно, трябва да го програмирате да се движи в произволна
посока след удара.
В Snap! посоките се изразяват в градуси:
0 градуса - НАГОРЕ
180 градуса - НАДОЛУ
90 градуса - ДЯСНО
270 градуса - НАЛЯВО
С други думи, ако изберете произволно числото от 0 до 3 и го
умножите по 90, ще получите произволна посока!
Той трябва да се движи, докато не се сблъска с Pacman. Тогава
играта свършва.
when clicked set direction = to 0 repeat until touching pacman ? point in direction direction move • steps if touching ? move • steps set direction = to pick random 0 to 3 × 90 say GAMEOVER! for 2 secs stop all *
[Стъпка 6]
Сега трябва да програмирате кога ще спечелите играта.
Това ще стане, когато съберете 20 звезди. Имате брояч на
звездите вътре в Растап скрипта. В началото го инициализирайте
на 0 и след това увеличавайте стойността му с 1 всеки път, когато
клонингът изпрати съобщение, че вие (като играч) сте го събрали.
Ако броячът стигне до 20, РАСМАN печели и играта свършва.





	when clicked go to x: 0 y: 0 set tooke to 0 set to 0 set to 0 set tooke tooke tooke to 0 set tooke to 0 set tooke to 0
Инструменти и	 Цялата дейност в Snap!:
ресурси за учителя	https://snap.berkeley.edu/project?user=zapusek&project=pac man_clone
	 Lajovic, S. (2011). Scratch. Nauči se programirati in postani računalniški maček. Ljubljana: Pasadena.
	 Vorderman, C. (2017). Računalniško programiranje za otroke. Ljubljana: MK.
Ресурси/материали	 Примерен шаблон в Snap!:
за ученика	https://snap.berkeley.edu/project?user=zapusek&project=pac
	man_template
	• Инструкции за ученика
	(C4G21_InstructionsForStudent_BG.docx)



ЛИТЕРАТУРА



Co-funded by the Erasmus+ Programme of the European Union

Дурева Д., М. Касева, Г. Тупаров, Компютърно моделиране, 4. клас, Просвета, 2018, София (Dureva, D., M. Kaseva, G. Tuparov, Kompyutarno modelirane, 4. klas, Prosveta, 2019, Sofia)

Lajovic, S. (2011). Scratch. Nauči se programirati in postani računalniški maček. Ljubljana: Pasadena.

Rugelj, J. (2019). Game design based learning of programming.

Vorderman, C. (2017). *Računalniško programiranje za otroke*. Ljubljana: MK.