**Learning Scenario 10 – Feeding the cats**

Task: Program the game in which the shelter keeper will repeatedly ask the player for the number of cats she can feed in a certain room. The number depends on the number (2 to 10) and size (2 to 5) of the bowls. For each room those two numbers have to be assigned randomly. The size of the bowl tells how many cats can eat from it, for example if bowl size is 3 that means 3 cats can eat from it.

We also have to have a counter that will count the right answers. Create a game in which the player will have to guess the right number of cats that we can feed in each room. After the activity give a feedback about how many times the player answered correctly and how many times she was wrong.

The process of creating the game:

1. Open the template file:

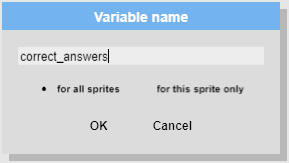
<https://snap.berkeley.edu/project?user=zapusek&project=cat_feeding_template>

There is an image for the background and the main character - cat shelter keeper.

|  |  |
| --- | --- |
|  |  |

1. We want to count the correct answers. If we want to store a value in a program we have to use a variable. We can access commands regarding variables in the “Variables” group. New variable is created when we click on: .

Variables assign a name to a value, so when we make a new one, we have to name it first. It is suggested that we use a mnemonic name, this means that we can tell what kind of value is stored in a variable from its name. For counting correct answers we can choose the name “correct\_answers”.



The value of the variable can be set to or changed by some value. If we want to set the value of the variable we use the block: . Everytime we set a value to a variable, the previous value is overwritten.

If we want to change the current value by some other value, we have to use block: . Current value of the variable will be changed by the value we specify in a white space.

1. Now we will start to actually code the game. First think about how activities in each room will actually be very similar, the same even. In each room we have to:
   1. assign a random value for the number of bowls and for the bowl size.

In the “Operators” group we can find the block that returns the random value from an interval that we can specify. For example:  will return the random number between 1 and 10.

We will need the values of the number and size of the bowls in each room stored for later when we will have to calculate the right answer. This can be achieved by creating new variables and assigning those values to them.

Create two variables:

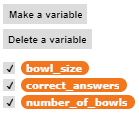
1. for the number of bowls in each room
2. for the size of the bowl in each room

Assign them random values according to the specification of this game. Use the

blocks:  and .

* 1. Player has to know which random values computer selected, so we have to inform her. Use blocks:  ,  and the values of the variables that store the number and size of bowls:  and .

You can find references to variables in the “Variables” group:



* 1. Now we have to prompt the player to input her answer. If we want to get the player input we have to use input block from “Sensing” group: .

When the player write her answer into input field it is stored in the “answer” variable: .

* 1. Next we have to check if the player’s answer is correct or not. Think how you can calculate the right answer from the number and size of the bowls. If we compare the value stored in  with the number of cats we can feed (we calculate this value), we can find out if the answer is correct or not.

Calculate the correct number of cats we can feed in each room using these blocks:

,  and .

* + 1. if the answer is correct: congratulate the player and add one to the value of the correct\_answer variable.
    2. if the answer is wrong: give the appropriate feedback.

We can differentiate between exactly two possible situations with the use of if-else block:

Med dvema situacijama (pravilen ali napačen odgovor) ločimo z uporabo bloka za pogojni stavek “če-sicer”.



If the answer will be correct it will be equal to the calculated value, otherwise it won’t be. Complete the code using the block that checks equality: .

1. We implemented the code for one room, but there are ten rooms in a shelter. We could copy the above code ten times and place it sequentially but that would not be an optimal way to do it. Instead of doing that we can use a loop that will repeat the same code ten times. The most simple loop for achieving that is repeat [n] times loop:



1. When the game is over you have to provide the feedback: number of correct and wrong answers. The number of correct answers is stored in a variable and the number of wrong answers can be calculated. If we know, that player will input the number 10 times and we also know how many times he answered correctly, we can find out how many times she was wrong. Complete the game using these blocks:
2. Na koncu izpiši število pravilnih in napačnih odgovorov. Število pravilnih je shranjeno v spremenljivki *tocke*. Napačne pa lahko izračunamo. Če vemo, da je možno zbrati največ deset točk, prav tako pa kolikokrat je igralec odgovoril pravilno, lahko število napačnih odgovorov izračunamo: ,  and .
3. You can upgrade the game using these suggestions:
   1. Shelter keeper includes the number of each room in her question. For example: “Guess the number of cats I can feed in room 5”.
   2. If the player's answer is wrong, she tells her the correct number.
   3. Instead of ten rooms, the room number is random or the player inputs the number of rooms at the beginning of the game.
   4. Add your own ideas and upgrade the game to your likings.