Python Course 2023 - Assignment

Torkil Thomsen

1 Python Assignemnt 3

Added comments are in bold or cursive to increase the understanding of the problems.

To reduce the possibility of software assistance or other helping tools, the assignment must be completed only with the tools from chapter 1 to 4.

1.1 Exercise 4.10 Guess the Number

Write a script that plays "guess the number." Choose the number to be guessed by selecting a random integer in the range 1 to 1000. Do not reveal this number to the user. Display the prompt "Guess my number between 1 and 1000 with the fewest guesses:". The player inputs a first guess. If the guess is incorrect, display "Too high. Try again." or "Too low. Try again." as appropriate to help the player "zero in" on the correct answer, then prompt the user for the next guess. When the user enters the correct answer, display "Congratulations. You guessed the number!", and allow the user to choose whether to play again.

This is a game, therefore it should be designed to run until the user wants to quit. A hint would to be use a sentinel value or a boolean flag to determine this logic. When working on the solution, print the assigned number to allow quick debugging.

1.2 Exercises 4.14-4.17 Computer Assisted Instruction

The following exercises are together, if you complete 4.17 in a single file, that is enough to hand in and pass the assignment. If you hand in all separate files, it is fine, but only the final product of the 4.17, will be assessed.

Part 1 to 4 is related to assignments from 4.14 to 4.17

Part 1. Computer assisted instruction (CAI) refers to the use of computers in education. Write a script to help an elementary school student learn multiplication. Create a function that randomly generates and returns a tuple of two positive one-digit integers. Use that function's result in your script to prompt the user with a question, such as:

How much is 6 times 7

For the correct answer, display "Very good!" and ask for another multiplication question. For an incorrect answer, display the message "No. Please try again." and let the student try the same question repeatedly until the student finally gets it right.

Part 2. Varying the computer's responses can help hold the student's attention. Modify the previous exercise so that various comments are displayed for each answer. The possible responses to a correct answer should include "Very good!", "Nice work!" and "Keep up the good work!". Possible responses to an incorrect answer should include "No. Please try again.", "Wrong. Try once more." and "No. Keep trying". Choose a number from 1 to 3, then use that value to select one of the three appropriate responses to each correct or incorrect answer.

Part 3. Modify the previous exercise to allow the user to enter a difficulty level. At a difficult level of 1, the program should use only single-digit numbers in the problems and at difficulty level of 2, numbers as large as two digits.

Please note that for difficulty 2*, it should also be possible to have single digit numbers* $0 - 99$*.*

If left unhandled division 0 can occur as a single digit, this would cause the program to crash, a hint would be to add some logic to handle if 0 occurs

Part 4. Modify the previous exercise to allow the user to pick a type of arithmetic problem to study - 1 means addition problems only, 2 means subtraction

problems only, 3 means multiplication problems only, 4 means division problems only **(Avoid dividing by 0)** and 5 means a random mixture of all these types.

It's ideal to have a game loop and as with exercise 4.10, some logic to control when the game should end. It's advisable to create a menu and having a sixth option of quitting the game necessary to close the program.