LC CS Python

Student Exercise Book

LEAVING CERTIFICATE
COMPUTER SCIENCE

Section 5

Programming Logic 1

If statements

Name: _____

Task 1

Study the illustration for the 'Hangman' game carefully and identify the areas where sequence, selection and iteration could be used in a programming solution for the game.

Task 2

Evaluate the following Boolean expressions

`True and False`
`not True or False`
`True and not True`
`not True or not False`
`not True and not False`

Task 3

```
1.  # A program to demonstrate the single if statement
2.  import random
3.
4.  number = random.randint(1, 10)
5.  # print(number)
6.
7.  guess = int(input("Enter a number between 1 and 10: "))
8.
9.  # Evaluate the condition
10. if guess == number:
11.     print("Your guess was correct")
12.     print("Well done!")
13.
14. print("Goodbye")
```

Guessing Game v1

(Uncommenting line 5 will help you test this program faster)

Type up and run the code. Test it fully. Log any questions you still have in relation to this code.

Task 4

Modify the code from Guessing Game V1 to Guessing Game V2 (code given below).

```
1.  # A program to demonstrate the double if statement
2.  import random
3.
4.  number = random.randint(1, 10)
5.  print(number) # comment this line out later!
6.
7.  guess = int(input("Enter a number between 1 and 10: "))
8.
9.  # Evaluate the condition
10. if guess == number:
11.     print("Your guess was correct")
12.     print("Well done!")
13.     print(" ..... play again soon!")
14. else:
15.     print("Hard luck!")
16.     print("Incorrect guess")
17.     print(" ..... play again soon!")
18.
19. print("Goodbye")
```

Guessing Game v2

Test the code fully. Log any questions you still have in relation to this code.

Task 5

Compare the logic of the two code snippets below. What do you notice?

```
# Evaluate the condition
if guess != number:
    print("Hard luck!")
    print("Incorrect guess")
else:
    print("Your guess was correct")
    print("Well done!")

print(" ..... play again soon!")
print("Goodbye")
```

```
# Evaluate the condition
if guess == number:
    print("Your guess was correct")
    print("Well done!")
    print(" ..... play again soon!")
else:
    print("Hard luck!")
    print("Incorrect guess")
    print(" ..... play again soon!")

print("Goodbye")
```

Task 6

Re order the individual lines of code shown below into a program that:

a) generates two random numbers between 0 and 12
b) calculates their product
c) prompts the user to enter the product of the two numbers
d) ***displays an appropriate response to the user's attempt***

```
print("Goodbye")
print("The correct answer was %d" %(n1*n2))
n2 = random.randint(0, 12)
else:
print("%d * %d" %(n1,n2))
print("Incorrect!")
ans = int(input("Enter your answer: "))
```

```
if ans == n1*n2:
ans = n1*n2
if ans = n1*n2:
import random
n1 = random.randint(0, 12)
print("Correct!")
n1*n2 = ans
```

Note - four of the lines are surplus to requirements.

Task 7

Modify the code from Guessing Game V2 to Guessing Game V3 (code given below).

```
1.  # A program to demonstrate the multiple if statement
2.  import random
3.
4.  number = random.randint(1, 10)
5.  print(number) # comment this line out later!
6.
7.  guess = int(input("Enter a number between 1 and 10: "))
8.
9.  # Evaluate the condition
10. if guess == number:
11.       print("Correct")
12.       print("Well done!")
13. elif guess < number:
14.     print("Hard luck!")
15.     print("Too low")
16. else:
17.     print("Hard luck!")
18.     print("Too high")
19.
20. print("Goodbye")
```

Type up and run the code. Test it fully. Log any questions you still have in relation to this code.

Task 8

Fill in the blanks below without altering the logic of the example program on the previous page. Log your thoughts as you proceed.