

Steffan Wilkins 2101242

assignment

by Steffan Wilkins

Submission date: 28-Apr-2022 09:06PM (UTC+0100)

Submission ID: 177712616

File name: 32671_Steffan_Wilkins_Steffan_Wilkins_2101242_assignment_410175_1153589098.docx (1.11M)

Word count: 1602

Character count: 7273

Part A

A1. A variable "is a value that can change, depending on conditions or on information passed to the program"[1]. Variables can be used in two main ways, one way is to have an input from the user which then with variables it can be used in multiple ways later in the code. The other way to use variables is for a calculation so that when using the print function, the code can be more organised to lower mistakes by having the calculation prior to the print.

A2. An if/else statement is used "used to execute both the true part and the false part of a given condition"[2]. If statements are used to make sure a part of code would only be executed if a certain criterion is meant. Elif statements can be used to make a joint statement in case of a different option of the if statement. Else statements are used for the remaining options for the if statement. For example, if you have an if the statement of an if statement which is ages between 18 and 30 you could have an else statement which would be the ages 18 and below and 30 and above.

A3. There are two types of loops while and for loops. "A for loop is used for iterating over a sequence"[3]. For loops, the developer will have to set how many times the loop will go around, for loops are useful when working with lists as with the counter you can display all variables in a list very easily or find something you want. While loops only "execute a set of statements as long as a condition is true"[4]. While loops are used when the developer does not know how many loops there will be, such as the number of retries the user needs in a guessing game.

B1. The following exercise will need three inputs name age and height for age and height there should be validation to make sure that someone doesn't input that they are 300 years old or 10 metres tall. The name will need to be a string. Age will need to be an integer. Height will have to be float.

```
aCorrect = 0
hCorrect = 0

Name = input("Please enter your name: ")

while aCorrect == 0: # This is to allow the user to retype there age if
they've done it wrong
    Age = int(input("Please enter your age: "))
    if 0 < Age < 150:
        aCorrect = 1 # This is to stop the code from looping once they have
entered an acceptable age

    while hCorrect == 0: # This is to allow the user to retype there
height if they've done it wrong
        Height = float(input("Please enter your height in metres: "))
        if 0.5 < Height < 3:
```

```

        hCorrect = 1 # This is to stop the code from looping once
        they have entered an acceptable age
        print(f"Thanks, you entered: {Name} - {Age} - {Height}m tall")
        # I've formatted the print so that there would be not gap
        between the height and the m
    else:
        print("The height you entered seems to be extreme")
else:
    print("The age you entered seems to be too high or too low")

```

The screenshot shows a Python IDE with the following code in the editor:

```

1 Name = input("Please enter your name: ")
2
3 while aCorrect == 0: # This is to allow the user to retype there age if they've done it wrong
4     Age = int(input("Please enter your age: "))
5     if 0 < Age < 150:
6         aCorrect = 1 # This is to stop the code from looping once they have entered an acceptable age
7
8     while hCorrect == 0: # This is to allow the user to retype there height if they've done it wrong
9         Height = float(input("Please enter your height in metres: "))
10        if 0.5 < Height < 3:
11            hCorrect = 1 # This is to stop the code from looping once they have entered an acceptable age
12            print(f"Thanks, you entered: {Name} - {Age} - {Height}m tall")
13            # I've formatted the print so that there would be not gap between the height and the m
14        else:
15            print("The height you entered seems to be extreme")
16    else:
17        print("The age you entered seems to be too high or too low")
18
19 while aCorrect == 0 : if 0 < Age < 150

```

The console output shows the following sequence of events:

```

C:\Users\steff\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/steff/PycharmProjects/pythonProject/main.py
Please enter your name: Steffan
Please enter your age: 19
Please enter your height in metres: 1.8
Thanks, you entered: Steffan - 19 - 1.8m tall
Process finished with exit code 0

```

The screenshot shows the same Python IDE with the same code. The console output shows the following sequence of events:

```

C:\Users\steff\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/steff/PycharmProjects/pythonProject/main.py
Please enter your name: Steffan
Please enter your age: 170
The age you entered seems to be too high or too low
Please enter your age: 19
Please enter your height in metres: 30.20
The height you entered seems to be extreme
Please enter your height in metres: 1.8
Thanks, you entered: Steffan - 19 - 1.8m tall
Process finished with exit code 0

```

B2. For this exercise, the user will have to input the length and height of the square and rectangle and then they have to input the radius of the circle. Then the code will take away the area of the square and rectangle from the area of the circle. There will need to be validations to check to see if the radius is the biggest dimension than any other variable. For this exercise, I will need formulas for the area of a circle and the area of a square and rectangle.

```
import math # This is to have an accurate number for pi later

sq = 9999999999 # This is to start the loop where the radius needs to be
bigger
rw = 9999999999
rl = 9999999999

radius = int(input("Please enter radius of the circle in cm: "))
while radius < sq:
    sq = int(input("Please insert the side of the square in cm: "))
    if radius < sq:
        print("That seems a little big, please try again:")
while radius < rw:
    rw = int(input("Please insert the width of the object in cm: "))
    if radius < rw:
        print("That seems a little big, please try again:")
while radius < rl:
    rl = int(input("Please insert the length of the object in cm: "))
    if radius < rl:
        print("That seems a little big, please try again:")
sa = sq ** 2
ra = rw * rl

shaded = round((math.pi * radius ** 2) - (sa + ra), 2) # This is to
calculate the shaded area
print(f"The shaded area is {shaded}cm^2")
```

```
main.py
1 r1 = 999999999
2
3
4
5
6
7
8 radius = int(input("Please enter radius of the circle in cm: "))
9 while radius < sq:
10     sq = int(input("Please insert the side of the square in cm: "))
11     if radius < sq:
12         print("That seems a little big, please try again!")
13
14 while radius < rw:
15     rw = int(input("Please insert the width of the object in cm: "))
16     if radius < rw:
17         print("That seems a little big, please try again!")
18
19 while radius < rl:
20     rl = int(input("Please insert the length of the object in cm: "))
21     if radius < rl:
22         print("That seems a little big, please try again!")
23
24 sa = sq ** 2
25 ra = rw * rl
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Run main.py

C:\Users\steff\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/steff/PycharmProjects/pythonProject/main.py

Please enter radius of the circle in cm: 10

Please insert the side of the square in cm: 5

Please insert the width of the object in cm: 5

Please insert the length of the object in cm: 10

The shaded area is 630.86cm²

Process finished with exit code 0

```
main.py
1 r1 = 999999999
2
3
4
5
6
7
8 radius = int(input("Please enter radius of the circle in cm: "))
9 while radius < sq:
10     sq = int(input("Please insert the side of the square in cm: "))
11     if radius < sq:
12         print("That seems a little big, please try again!")
13
14 while radius < rw:
15     rw = int(input("Please insert the width of the object in cm: "))
16     if radius < rw:
17         print("That seems a little big, please try again!")
18
19 while radius < rl:
20     rl = int(input("Please insert the length of the object in cm: "))
21     if radius < rl:
22         print("That seems a little big, please try again!")
23
24 sa = sq ** 2
25 ra = rw * rl
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Run main.py

C:\Users\steff\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/steff/PycharmProjects/pythonProject/main.py

Please enter radius of the circle in cm: 10

Please insert the side of the square in cm: 5

That seems a little big, please try again:

Please insert the width of the object in cm: 5

Please insert the length of the object in cm: 5

The shaded area is 246.16cm²

Process finished with exit code 0

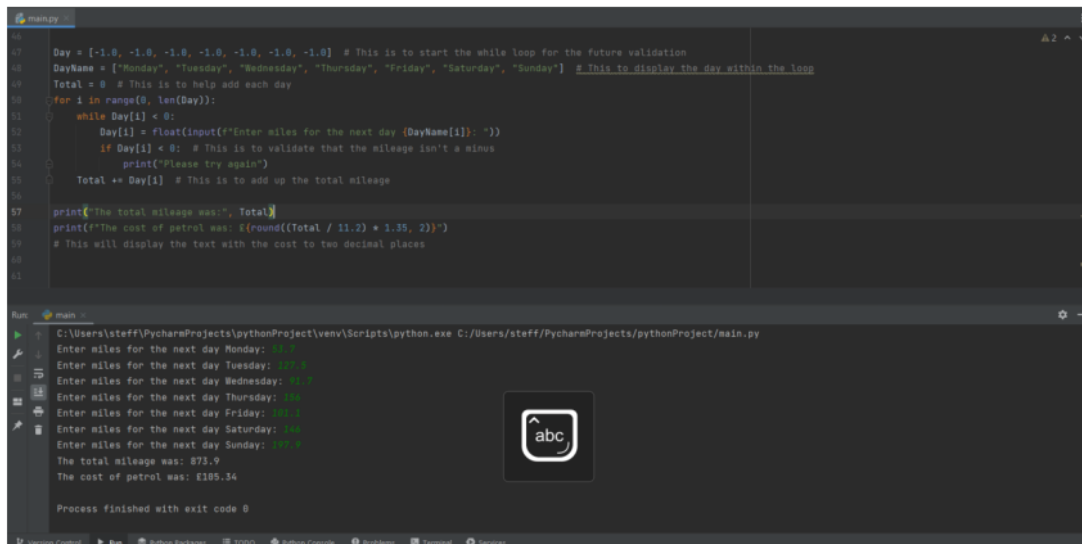
B3. The user will have to insert the miles for each day of the week then the code will display the total millage, and it will calculate how much petrol was spent. In this, I will need to make a validation check so that the user can input a number that can't work such as a minus number of miles in a day.

```

Day = [-1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0] # This is to start the
while loop for the future validation
DayName = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday",
"Saturday", "Sunday"] # This to display the day within the loop
Total = 0 # This is to help add each day
for i in range(0, len(Day)):
    while Day[i] < 0:
        Day[i] = float(input(f"Enter miles for the next day {DayName[i]}: "))
        if Day[i] < 0: # This is to validate that the mileage isn't a minus
            print("Please try again")
        Total += Day[i] # This is to add up the total mileage

print("The total mileage was:", Total)
print(f"The cost of petrol was: £{round((Total / 11.2) * 1.35, 2)}")
# This will display the text with the cost to two decimal places

```



The screenshot shows the PyCharm IDE with a Python script in the editor and its execution output in the Run console. The script calculates the total mileage for a week and the cost of petrol. The output shows the user entering miles for each day, the total mileage, and the cost of petrol.

```

C:\Users\steff\PycharmProjects\pythonProject\venv\Scripts\python.exe C:\Users\steff\PycharmProjects\pythonProject/main.py
Enter miles for the next day Monday: 80
Enter miles for the next day Tuesday: 80
Enter miles for the next day Wednesday: 80
Enter miles for the next day Thursday: 80
Enter miles for the next day Friday: 80
Enter miles for the next day Saturday: 80
Enter miles for the next day Sunday: 80
The total mileage was: 560
The cost of petrol was: £105.34
Process finished with exit code 0

```

```

16
17 Day = [-1.0, -1.0, -1.0, -1.0, -1.0, -1.0, -1.0] # This is to start the while loop for the future validation
18 DayName = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"] # This is to display the day within the loop
19 Total = 0 # This is to help add each day
20 for i in range(0, len(Day)):
21     while Day[i] < 0:
22         Day[i] = float(input("Enter miles for the next day (DayName[i]): "))
23         if Day[i] < 0: # This is to validate that the mileage isn't a minus
24             print("Please try again")
25         Total += Day[i] # This is to add up the total mileage
26
27 print("The total mileage was:", Total)
28 print(f"The cost of petrol was: £(round((Total / 11.2) * 1.25, 2))")
29 # This will display the text with the cost to two decimal places
30
31

```

Run: C:\Users\steff\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/steff/PycharmProjects/pythonProject/main.py

```

Enter miles for the next day Monday: 10
Enter miles for the next day Tuesday: 10
Please try again
Enter miles for the next day Tuesday: 1
Enter miles for the next day Wednesday: 10
Enter miles for the next day Thursday: 10
Enter miles for the next day Friday: 10
Enter miles for the next day Saturday: 10
Enter miles for the next day Sunday: 10
The total mileage was: 559.6
The cost of petrol was: £67.45

```

B4. The plan for this exercise is to assign 6 animals in 12 different positions with them in 2 locations each. The user will be able to pick two numbers and the computer will tell them if they got a pair or not. If not, the computer will display the two animals to give them a hint for future guesses. The user will be allowed to fail at the most 5 failures. When all have been paired then the game will end. For validations, the user can't input outside of the range and the user can't input the same number twice.

```
import random # This allows me to use the random function
```

```
def update(): # this is to not needing to put this in two places later
    print("1 2 3 4 5 6 7 8 9 10 11 12")
    for i in range(0, len(unknown)): # This is to print off the current
        paired positions
        print(unknown[i], end=" ")
    print("")
```

```
Animals = ["Dog", "Cat", "Horse", "Frog", "Penguin", "Mouse"]
order = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

```
for i in range(0, len(Animals)): # This loop sets 6 animals in 12 different
    slots twice each at random
    work = 0
    while not (work == 2):
        slot = random.randint(0, 11)
        if order[slot] == 0:
            order[slot] = Animals[i]
```

```

        work += 1

unknown = ["?", "?", "?", "?", "?", "?", "?", "?", "?", "?", "?", "?"]

print("Welcome to the pairs game! There are 6 pairs to find. \nHere is the
board:")
update()
fail = 0 # This will be used to tally the amount of failures

while fail < 5:
    cheating = 1
    while cheating == 1: # This is to stop the user for entering the same
number twice
        inRange1 = 0
        inRange2 = 0
        while inRange1 == 0: # This is for validation to make sure the number
entered is between 1 and 12
            guess1 = int(input("\nPlease enter guess 1: ")) - 1 # This makes
it normal for people to insert numbers but make sense to the computer and the
lists
            if -1 < guess1 < 12:
                inRange1 = 1
            else:
                print("Number not in range please try again")
        while inRange2 == 0:
            guess2 = int(input("Please enter guess 2: ")) - 1
            if -1 < guess1 < 12 and not (guess1 == guess2):
                inRange2 = 1
            else:
                print("Number not in range please try again")
        if guess1 == guess2:
            print("You can't input the same number twice")
        else:
            cheating = 0
        if order[guess1] == order[guess2]: # This is to check if the animals are
the same
            print(f"well done, that's a match on {order[guess1]}!")
            unknown[guess1] = "x" # This marks the spots to make it easier for
the player
            unknown[guess2] = "x"
            update()
        else:
            fail += 1 # This is to mark a failure
            print(f"Not a match. you have {5 - fail} tries left.")
            print(f"Position {guess1 + 1} was {order[guess1]} and position {guess2
+ 1} was {order[guess2]}")
            correct = 0
            for i in range(0, len(unknown)): # This is to check how many the user has
gotten correct if all are done the computer will end the game

```

```

        if unknown[i] == "x":
            correct += 1
    if correct == 12:
        fail = 10

if fail == 5: # This is to make the user aware they lost
    print("Out of tries. Good luck next time")
else: # This is to congratulate the user
    print("Well done! You won you deserve a cookie.")

```

```

main.py
while fail < 5 : while cheating == 1 : while inRange1 == 0

C:\Users\steff\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/steff/PycharmProjects/pythonProject/main.py
Welcome to the pairs game! There are 6 pairs to find.
Here is the board:
1 2 3 4 5 6 7 8 9 10 11 12
? ? ? ? ? ? ? ? ? ? ? ?

Please enter guess 1: 1
Please enter guess 2: 1
You can't input the same number twice

Please enter guess 1: 10
Number not in range please try again

Please enter guess 1:

```

```

main.py
while fail < 5 : while cheating == 1 : while inRange1 == 0

well done, that's a match on Horse!
1 2 3 4 5 6 7 8 9 10 11 12
? ? x x x x ? ? ? ? ? ?

Please enter guess 1: 7
Please enter guess 2: 8
Not a match, you have 1 tries left.
Position 7 was Dog and position 8 was Mouse

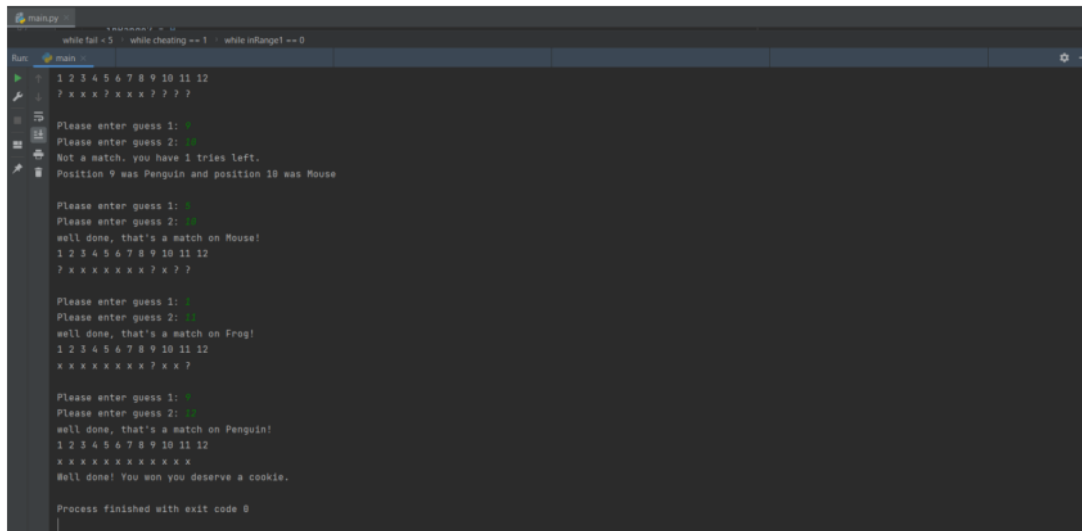
Please enter guess 1: 7
Please enter guess 2: 9
well done, that's a match on Mouse!
1 2 3 4 5 6 7 8 9 10 11 12
x ? x x x x ? ? ? ? ? ?

Please enter guess 1: 7
Please enter guess 2: 7
well done, that's a match on Dog!
1 2 3 4 5 6 7 8 9 10 11 12
x x x x x x x ? ? ? ? ?

Please enter guess 1: 9
Please enter guess 2: 11
Not a match, you have 0 tries left.
Position 9 was Cat and position 11 was Penguin
Out of tries. Good luck next time

Process finished with exit code 0

```



```
while fail < 5 : while cheating == 1 : while inflange1 == 0 :
run main
1 2 3 4 5 6 7 8 9 10 11 12
? x x ? x x ? ? ? ?
Please enter guess 1: 4
Please enter guess 2: 10
Not a match, you have 1 tries left.
Position 9 was Penguin and position 10 was Mouse
Please enter guess 1: 4
Please enter guess 2: 10
well done, that's a match on Mouse!
1 2 3 4 5 6 7 8 9 10 11 12
? x x x x x ? x ? ?
Please enter guess 1: 4
Please enter guess 2: 10
well done, that's a match on Frog!
1 2 3 4 5 6 7 8 9 10 11 12
x x x x x x ? x x ?
Please enter guess 1: 4
Please enter guess 2: 10
well done, that's a match on Penguin!
1 2 3 4 5 6 7 8 9 10 11 12
x x x x x x x x x x
Well done! You won you deserve a cookie.
Process finished with exit code 0
```

- [1] Tech Target Contributor, "variable", *WhatIs.com*, Oct. 2021. [Online] Available: <https://www.techtarget.com/whatis/definition/variable#:~:text=In%20programming%2C%20a%20variable%20is,uses%20when%20it%20is%20running.> [Accessed Apr. 04, 2022].
- [2] A.Biswal, "Understanding Python If-Else Statement", *simplilearn*, Feb. 7, 2022. [Online] Available: <https://www.simplilearn.com/tutorials/python-tutorial/python-if-else-statement#:~:text=The%20if%2Delse%20statement%20is,else%20block%20code%20is%20executed.> [Accessed Apr. 04, 2022].
- [3] "Python While Loops", *W3schools*, [Online] Available: https://www.w3schools.com/python/python_for_loops.asp [Accessed Apr. 04, 2022].
- [4] "Python While Loops", *W3schools*, [Online] Available: https://www.w3schools.com/python/python_while_loops.asp [Accessed Apr. 04, 2022].

Steffan Wilkins 2101242 assignment

GRADEMARK REPORT

FINAL GRADE

76/100

GENERAL COMMENTS

Instructor

PART A

Variables - Quite simply, variables are used to store values within a computer program. You have talked about variables, but you haven't talked about types - you need to say what the types are ie int, float string. You also need to talk about the specific way that Python infers the variable type and how/why these types are considered dynamic (in Python, types can change within the lifetime of the program, for example, when dividing two integers, the resulting type is changed to float) 1.5/5

If/Else -If/Else - You've got the main point but you also need to talk about multiple conditions that have two pieces of logic (using and/or) I think you're trying to say this in your last sentence, but the expression is a bit confused. No code examples? 3/5

Loops - You've got the main points but no code examples? 3.5/5

PART B

As a general comment I don't think you have consulted the accompanying document "How to lay out Part B of your Assignment"? - it's on Moodle and I also went through it in class/on my video.

Exercise 1

Logical Stages-Testing - Logical stages are sort of shown, but they need to be numbered and state what happens if someone did input they were 300 years old (and what the acceptable limits are). This is so a programmer can correctly implement the code. Testing is there, but you need to crop to just the output and state what you're testing and whether the program behaved as expected (again, see handout for this) 2.5/5

Coding-Annotation - Working program with thorough validation. 10/10

Exercise 2

Logical Stages-Testing - Logical stages need to be presented numerically.(Same comments as above) 2.5/5

Coding-Annotation - Good program, you consider size of circle against the square/rectangle, could also consider size of square/rectangle against each other. Not sure why you've gone for ints - floats would be more appropriate. 8.5 /10

Exercise 3

Logical Stages-Testing - Same comments as above. 6/10

Coding-Annotation - A good attempt with some use of lists/loops. You could also have validated on upper limit of mileage 9/10

Exercise 4

Logical Stages-Testing - Same comments as above
6/10

Coding-Annotation - Good effort. Use of functions
could have been more extensive. Validation is good.
Fully working game 23/25

General comment: All assignments should really have
a short Introduction and a conclusion - just to set the
scene on what the assignment is about and in the
conclusion, to discuss how things went and what you
would do differently next time. There are no marks
assigned for that in this particular assignment, but it is
good academic practice.

Put your name on the top as well and number the
pages :)

Some good code here, you came down a bit in the
documentation, mainly the logical stages.

Assignment overall: 75.5%

Test mark: 95%

OVERALL MARK FOR MODULE: 81%

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9
