A red and blue text on a black background

Description automatically generated

## Assignment Title Page

|  |  |
| --- | --- |
| **STUDENT NAME:** | Angela Liu Qiyu |
| **P.I. NO.:** | J2181353 |
| **TITLE:** | TMA01 |
| **COURSE CODE:** | ANL252 |
| **TG:** | T03 |
| **SUBMISSION DATE:** | 14th September 2023 |

**Plagiarism and Collusion**

**Plagiarism** is the act of using or passing as one’s own, the ideas or writings of another without acknowledging or crediting the source from which the ideas are taken from.

**Collusion** is the act of submitting any academic work (including assignment, project or report) that was completed by another person and pass these work off as one’s own.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
| **Declaration** | | | | |
|  | | | | |
| I declare that this assignment is my own work, unless otherwise acknowledge or credited by appropriate referencing. I have read and abide by the SUSS Honour Code and I am aware of the penalties associated with plagiarism and collusion listed in the Student Handbook. | | | | |
|  | | | | |
| Initial: | Angela |  | Date: | 14th September 2023 |
|  | | | | |

**Question 1**

**(a)**

Plagiarism in coding happens when a student utilizes source code created by another individual and neglects to appropriately credit it, whether it's done on purpose or by accident, resulting in the submission of the code as their own (Cosma & Joy, 2008). This might be because the student is unfamiliar with the concepts, especially when they are new to coding. Some might not even know that copying other’s codes online is an offence of plagiarism.

There are a few ways to avoid plagiarism in coding, including:

* Acquire the knowledge of when and in what manner to incorporate external source code (Copyleaks, 2021).
* It is also important to “document your references” and cite the sources that you use that are created by others (Copyleaks, 2021). In the citation, the full information about the source, such as the author and authors’ names and the name of the website must be included.
* Try understanding the code and rewrite or recode it yourself (Copyleaks, 2021).
* Utilize software that examines for similarities such as turnitin (Lee, 2020).

**176 words – including in-text citations**

**(b)**

#Let's start the game! A number will be generated for you to guess~

import random

r = random.randint(0, 10)

print(f'Magic number has been generated')

print(f'You only have 2 chances to pick the correct number!')

success\_tries\_num = 0

input\_str = str(input("Please input your first try: "))

num1 = int(input\_str)

if num1 > r:

print(f"The number {num1} you have inputted is larger than the MAGIC NUMBER!")

elif num1 < r:

print(f"The number {num1} you have inputted is smaller than the MAGIC NUMBER!")

else:

print(f"BINGO! You have guessed the correct number {r}")

success\_tries\_num = 1

if success\_tries\_num == 0:

input\_str2 = str(input("Please input your second try: "))

num2 = int(input\_str2)

if num2 > r:

print(f"The number {num2} you have inputted is larger than the MAGIC NUMBER!")

elif num2 < r:

print(f"The number {num2} you have inputted is smaller than the MAGIC NUMBER!")

else:

print(f"BINGO! You have guessed the correct number {r}")

success\_tries\_num = 2

if success\_tries\_num > 0:

print(f"You have guessed the correct number with {success\_tries\_num} tries")

else:

print(f"You have failed to guess the MAGIC number {r}!")

Output:

Magic number has been generated

You only have 2 chances to pick the correct number!

Please input your first try: 5

The number 5 you have inputted is larger than the MAGIC NUMBER!

Please input your second try: 3

The number 3 you have inputted is larger than the MAGIC NUMBER!

You have failed to guess the MAGIC number 1!

Code: written by self

The code above is a mini game where the players have 2 tries to guess the correct number generated by the system, which in this case is Python. Every new round, Python will generate a new random number within 1 to 10. The player will need to input his first guess of the number generated by Python into this string: str(input("Please input your first try: ")). When the number has been inputted, Python will indicate whether the player has chosen the correct number. If the wrong number has been inputted, whether it is larger or smaller than the random number picked by Python, Python will tell the player that the number picked is wrong and the player will have another try to get the correct number. The game will end on the second round regardless of whether the correct number is chosen but the system will still indicate whether the number chosen is the correct one.

**156 words**

**(c)**

Code:

#Let's start the game! A number will be generated for you to guess~

import random

r = random.randint(0, 100)

print(f'Magic number has been generated')

# To define the number of tries

int\_number\_of\_tries = 15

print(f'You only have {int\_number\_of\_tries} chances to pick the correct number!')

# initialize the variable

i = 0

j = int\_number\_of\_tries - i

# To define exit flag

exit\_flag = 0

# while loop from i = 1 to 5

while i < int\_number\_of\_tries and exit\_flag == 0:

try:

i = i + 1

print(f"{i} try")

j = int\_number\_of\_tries - i

num = int(str(input(f"Please input your try: ")))

if num == 0:

print(f"You have decided to end the game after {i} number of failures.")

exit\_flag = 1

elif num > r:

print(f"The number {num} you have inputted is larger than the MAGIC NUMBER!")

elif num < r:

print(f"The number {num} you have inputted is smaller than the MAGIC NUMBER!")

else:

print(f"BINGO! You have guessed the correct number {r} by {i} times of tries")

exit\_flag = 1

except ValueError:

print(f"Oops! You have inputted a whole number! You have {j} tries left...")

Output:

Magic number has been generated

You only have 15 chances to pick the correct number!

1 try

Please input your try: 5

The number 5 inputted is smaller than the MAGIC NUMBER!

2 try

Please input your try: 3

The number 3 inputted is smaller than the MAGIC NUMBER!

3 try

Please input your try: 67

The number 67 inputted is larger than the MAGIC NUMBER!

4 try

Please input your try: 44

The number 44 inputted is smaller than the MAGIC NUMBER!

5 try

Please input your try: 50

The number 50 inputted is larger than the MAGIC NUMBER!

6 try

Please input your try: 60

The number 60 inputted is larger than the MAGIC NUMBER!

7 try

Please input your try: 0

You have decided to end the game after 7 number of failures.

The rationale behind rewriting the code is as follows:

* The code in the format as shown in part (c) is clearer than that in part (b) as a while loop is being used instead of if elif and else. By using a while loop, there will be fewer mistakes made as compared to the if elif and else as the number of lines being used in a while loop is relatively fewer than if elif and else if 15 tries is given to the player as shown in part (c). As the number of repeated lines increase, the risk of human error will also increase.
* The code at the end in part (c) also helps to capture exception, ValueError in Python, to indicate to the player than a non-integer has been inputted so that the player will have to input an integer so as not to waste his or her number of tries to guess the correct number generated.
* By adding num == 0, it gives the player a choice to end the game if they decided to stop the game halfway instead of having completing 15 number of tries, especially if the number of tries can be large.
* Additionally, different from part (b), in part (c), when the player has guessed the correct number that has been randomly generated by the system, the game will end, notifying the player that they have guessed the correct answer. Where in the previous code, there the game will continue even if the player has guessed the correct number.

Hence, the codes in part (c) are more meaningful than that in part (b) as this is more efficient for game developers to help reduce error, remove redundancy and is clearer.

**291 words**

**Question 2**

The first improvement I will do to the code given in Appendix 1 is to use dictionaries, changing the list. With dictionaries, the shop can fix the price of the products at the start and not wait for the customer to input the price as there is a risk of understatement of the price of the product.

The second improvement I will do is to initialise the total cost of the shopping cart as zero and according to what product the customer wants, the cost of the shopping cart will vary. Then I will use for loop to add the price of the products chosen together to achieve the total cost which will be easier for the customer to know how much they will need to pay in total.

The third improvement I will do is to print out the shopping cart list, the price of each individual product and the total cost for the entire shopping cart. This will make it clearer for the customer to see and can then confirm the products that they have chosen.

**178 words**

Edited code:

products = {

'laptop': 1200,

'mouse': 80,

'keyboard': 50,

'speaker': 50

}

query = 'yes'

cost = 0

updated\_items = []

print(f'We have a list of products here: {products}')

while query == 'yes':

item = str(input("Hello! What product do you want to buy? "))

if item not in products:

print('Wrong product! Please try again')

continue

updated\_items.append(item)

query = str(input("Would you like to continue? (yes/no) "))

print('This is your updated shopping cart:')

for i in range(len(updated\_items)):

item\_name = updated\_items[i]

item\_cost = products.get(item\_name)

cost += item\_cost

print(f'{i + 1}. {item\_name}: {item\_cost}')

print(f'Total cost: {cost}')

Output:

We have a list of products here: {'laptop': 1200, 'mouse': 80, 'keyboard': 50, 'speaker': 50}

Hello! What product do you want to buy? mouse

Would you like to continue? (yes/no) yes

Hello! What product do you want to buy? speaker

Would you like to continue? (yes/no) no

This is your updated shopping cart:

1. mouse: 80

2. speaker: 50

Total cost: 130

**Bibliography**

Plagiarism Online. (2010, Oct 19) *Source Code Plagiarism*. Retrieved from Plagiarism Online:

<http://web.science.mq.edu.au/~mtaylor/ponline/index.php?id=source-code-plagiarism#:~:text=Source%2Dcode%20plagiarism%20in%20programming,as%20his%2Fher%20own%20work>.

Copyleaks. (2021, Jul 9). *All You Need to Know About Plagiarism in Coding*. Retrieved from Copyleaks:

<https://copyleaks.com/blog/all-you-need-to-know-about-plagiarism-in-coding>

Lee, C. (2020, Jul 28). *What is Programming Plagiarism? Why Is It on the Rise?* Retrieved from Turnitin:

<https://www.turnitin.com/blog/what-is-programming-plagiarism-why-is-it-on-the-rise>