CSC1024 Programming Principles

Programming Project: Master Mind Computer Game

Presenter: Choon Han Wen Benjamin
Student ID: 22018345

Date: 11/7/2022
Video Link: https://youtu.be/pTgh110opQQ

Code Demonstration

Display Data

```
| ==================== |
|                      |
|  M A S T E R M I N D |
|                      |
| ==================== |

Welcome Player! This is Mastermind, a simple code breaking game.

How to play:
1. A random sequence of 4 fruits will be generated from a list of 4 fruits (Apple, Orange, Grape, Banana).

2. The player shall guess the generated fruit sequence, after guessing, the player will receive hints based on their guesses.
   The player will receive hints on:
   - Number of correct fruits in the correct positions.
   - Number of correct fruits but in the incorrect position.

3. The game will go on till the player guesses the correct sequence, the number of attempts will also be displayed at the end.

4. During entering guesses, take note to only enter the correct fruit types with no extra spacing.

5. If the player wishes to exit the game, please enter "exit" during the guessing phase.

It is as simple as it sounds! Have fun guessing!
```

- Welcome and inform the player the rules of the game

```
print('================================')
print('Game terminated, Have a nice day!')
print('================================')
```

```
Please enter guess for position 1: exit
Are you sure to exit? [Y/N]: y
================================
Game terminated, Have a nice day!
================================
PS C:\Users\Asus>
```

- Inform the player when the game has been terminated

```
Number of correct fruits in the correct position:  1
Number of correct fruits in the wrong position:  0
Attempts:  1
```

- Inform the player of their guesses and their number of attempts

```
print('****************************************')
print('****************************************')
print('**                                    **')
print('** Congragulations! You got it right! **')
print('**       You are the MASTERMIND       **')
print('**       You took ',count,' attempts        **')
print('**                                    **')
print('****************************************')
print('****************************************')
```

```
****************************************
****************************************
**                                    **
** Congragulations! You got it right! **
**       You are the MASTERMIND       **
**       You took  2  attempts        **
**                                    **
****************************************
****************************************
```

- Congratulate the player when their guess is correct

Input Data

```
guess_1 = str.capitalize(input('Please enter guess for position 1: '))
guess_2 = str.capitalize(input('Please enter guess for position 2: '))
guess_3 = str.capitalize(input('Please enter guess for position 3: '))
guess_4 = str.capitalize(input('Please enter guess for position 4: '))
```

```
Please enter guess for position 1:
```

- Input for player to enter their guess each round

```
exit_game = input('Are you sure to exit? [Y/N]')
```

- Input asking player whether they want to exit the game

```
if again == 'y' or again == 'Y':
    restart = True
    fruit_list_random = [random.choice(fruit_list), random.choice(fruit_list), random.choice(fruit_list), random.choice(fruit_list)]
```

- Input asking player whether they want to play again or exit the game

```
Please enter guess for position 1: exit
Are you sure to exit? [Y/N]:
```

```
Play again? Y/N:
```

Lists

```
fruit_list = ['Apple', 'Orange', 'Grape', 'Banana']
```

- Default list for all fruit types available

```
fruit_list_random = [random.choice(fruit_list), random.choice(fruit_list), random.choice(fruit_list), random.choice(fruit_list)]
```

- List that includes randomly generated sequence

```
guess_list = [guess_1, guess_2, guess_3, guess_4]
```

- List that includes player input

```
check_1 = []
check_2 = []
```

- Lists created to check user input

Random Choice From a List

```
import random
```

- Import random module

```
random.choice(fruit_list)
```

- Random function is called to generate a random selection

```
fruit_list_random = [random.choice(fruit_list), random.choice(fruit_list), random.choice(fruit_list), random.choice(fruit_list)]
```

If Statements

```
Input guesses below:
Available inputs: [Apple, Orange, Grape, Banana]

Please enter guess for position 1: apple
Please enter guess for position 2: orange
Please enter guess for position 3: wrong input
Please enter guess for position 4: grape

============================================================

////////////////////////////////////////////////////////////////////////////////////////////////////////////
/Invalid input detected, please make sure to input the correct fruit types (Apple, Orange, Grape or Banana)/
/                          Also take note to not enter extra spacing                                       /
////////////////////////////////////////////////////////////////////////////////////////////////////////////

============================================================

Input guesses below:
Available inputs: [Apple, Orange, Grape, Banana]

Please enter guess for position 1:
```

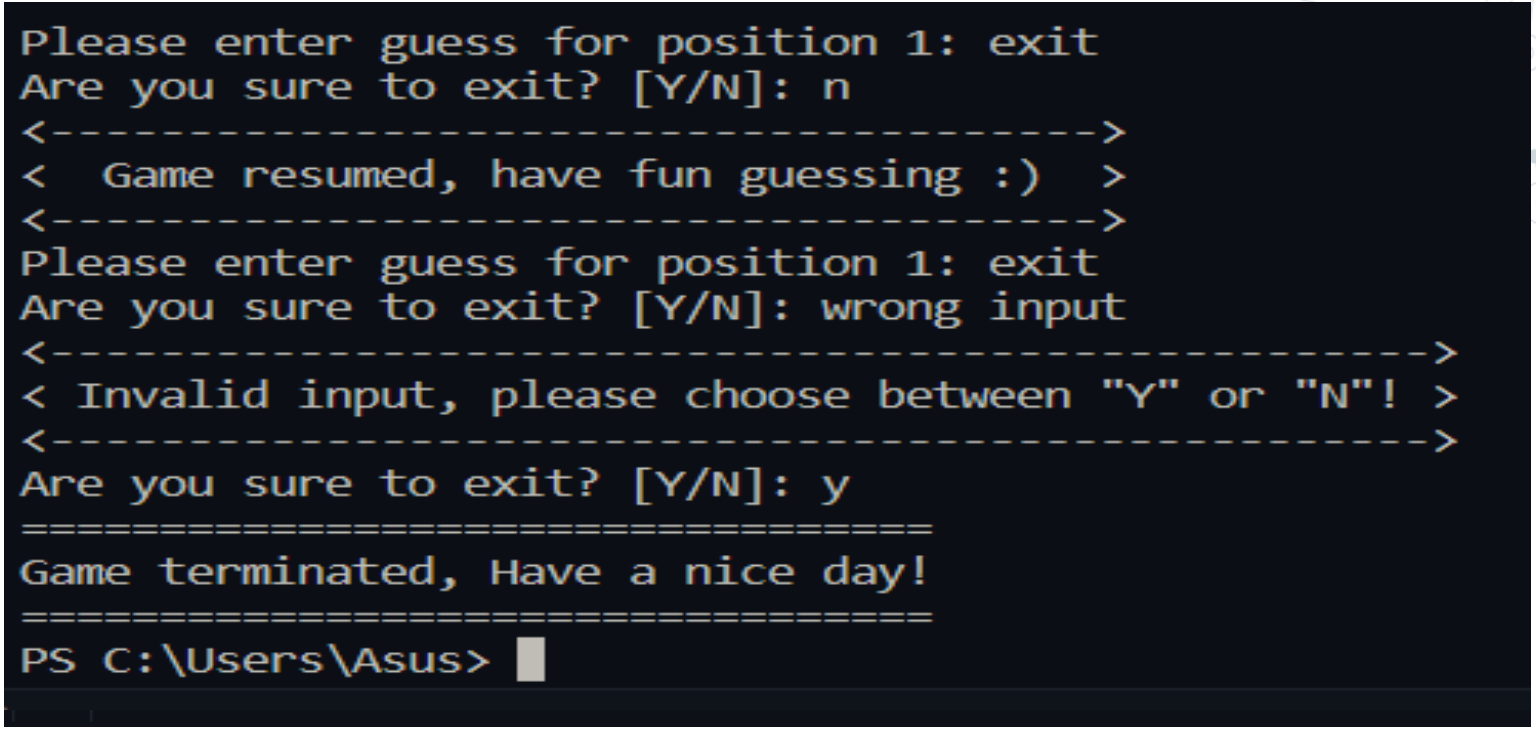
```
if position_correct_type_corect == 4 and position_wrong_type_correct == 0:
```

```
Your guesses:
[Position 1:  Grape  ]
[Position 2:  Grape  ]
[Position 3:  Banana  ]
[Position 4:  Banana  ]


Number of correct fruits in the correct position:  4
Number of correct fruits in the wrong position:  0
Attempts:  3

Sample answer:  ['Grape', 'Grape', 'Banana', 'Banana']
**********************************************
**********************************************
**                                          **
** Congragulations! You got it right!       **
**        You are the MASTERMIND            **
**        You took  3  attempts             **
**                                          **
**********************************************
**********************************************
Play again? Y/N:
```

```
m.choice(fruit_list)]
```



- If statement is used to check for the keyword ‘exit’
- Additional if statement is used within the original if statement to check if the player’s input, making it a nested if statement

Relational Operators

```
if guess_1 == 'Exit':
    exit_game = input('Are you sure to exit? [Y/N]: ')
    if exit_game == 'Y' or exit_game == 'y':
        end = ed()
        quit()
    elif exit_game == 'n' or exit_game == 'N':
        print('<---------------------------------------->')
        print('<  Game resumed, have fun guessing :)  >')
        print('<---------------------------------------->')
        guess_1 = str.capitalize(input('Please enter guess for position 1: '))
        break
    else:
        print('<------------------------------------------------>')
        print('< Invalid input, please choose between "Y" or "N"! >')
        print('<------------------------------------------------>')
        continue
```

- Only when the player's input is 'Exit' will the code ask the player if they want to exit the game
- If the player's input is 'Exit', the code will then check the next input for 'Y' or 'N', other inputs will be determined as invalid inputs
- If player inputs 'Y', the game will be terminated, 'N' will resume the game

```
if guess_list[i] == fruit_list_random[i]:
    position_correct_type_corect = position_correct_type_corect + 1
```

- If the player's input corresponds to the randomly generated fruit list, the counter for 'correct position and correct type' will increase

```
if position_correct_type_corect == 4 and position_wrong_type_correct == 0:
    print('**********************************')
    print('')
    print('Congragulations! You got it right!')
    print('    You are the MASTERMIND       ')
    print('    You took ',count,' attempts    ')
    print('')
    print('**********************************')
```

- If the counter for 'correct position and correct type' is four and the counter for 'wrong position and correct type' is zero, the code will display the winning message

Logical Operators

```
============================================================

Input guesses below:
Available inputs: [Apple, Orange, Grape, Banana]

Please enter guess for position 1: apple
Please enter guess for position 2: apple
Please enter guess for position 3: apple
Please enter guess for position 4: watermelon

============================================================

////////////////////////////////////////////////////////////////////////////////////////////////////////////
/Invalid input detected, please make sure to input the correct fruit types (Apple, Orange, Grape or Banana)/
/                          Also take note to not enter extra spacing                                       /
////////////////////////////////////////////////////////////////////////////////////////////////////////////

============================================================

Input guesses below:
Available inputs: [Apple, Orange, Grape, Banana]

Please enter guess for position 1:
```

Loops

```
Your guesses:
[Position 1:  Apple  ]
[Position 2:  Banana  ]
[Position 3:  Banana  ]
[Position 4:  Apple  ]



Number of correct fruits in the correct position:  4
Number of correct fruits in the wrong position:  0
Attempts:  2

Sample answer:  ['Apple', 'Banana', 'Banana', 'Apple']
****************************************
****************************************
**                                    **
** Congragulations! You got it right! **
**       You are the MASTERMIND       **
**       You took  2  attempts        **
**                                    **
****************************************
****************************************
Play again? Y/N: y
============================================================

Input guesses below:
Available inputs: [Apple, Orange, Grape, Banana]

Please enter guess for position 1:
```

- For Loop

```
check_1 = []
check_2 = []

i = 0
j = 0
position_correct_type_corect = 0
position_wrong_type_correct = 0

for i in range(len(guess_list)):
    if guess_list[i] == fruit_list_random[i]:
        position_correct_type_corect = position_correct_type_corect + 1
    else:
        check_1.append(guess_list[i])
        check_2.append(fruit_list_random[i])
while j < (len(check_1)):
    t = 0
    while t < (len(check_2)):
        if check_1[j] == check_2[t]:
            check_2.pop(t)
            position_wrong_type_correct = position_wrong_type_correct + 1
            break
        else:
            t = t + 1
    j = j + 1
```

- For and While loops are used in conjunction to check guess_list with fruit_list_random when conditions are fulfilled
- When conditions are met the code will stop looping

User Defined Functions

```
| ================== |
|                    |
|  M A S T E R M I N D |
|                    |
| ================== |

Welcome Player! This is Mastermind, a simple code breaking game.

How to play:
1. A random sequence of 4 fruits will be generated from a list of 4 fruits (Apple, Orange, Grape, Banana).

2. The player shall guess the correct sequence generated, with each guess, the player will be prompted on their guesses.
   The player will only receive prompts on:
   - Number of correct fruits in the correct positions.
   - Number of correct fruits but in the incorrect position.

3. The game will go on till the player gets a correct guess, the number of attempts taken will also be displayed at the end.

4. During entering guesses, take note to only enter the correct fruit types with no extra spaces.
```

```
Please enter guess for position 1: exit
Are you sure to exit? [Y/N]y
==================================
Game terminated, Have a nice day!
==================================
PS C:\Users\Asus>
```

Thanks For Listening

Any questions?

Feel free to comment below