

|  |
| --- |
| Assignment #  Winter-2025 |
|  |
| February 16th  Course Title: Programming Principles  Course Code: PROG10004  Authored by:  Student Name: Muhammad Mukry Student Number: 991798855 |

Contents

[Assignment 2 3](#_Toc190647720)

[Question #: 4](#_Toc190647721)

[Assignment1\_Exercises 4](#_Toc190647722)

[1. Use of Random and Math Module: 4](#_Toc190647723)

[2. Use of Arithmetic Operators and Git repositories 4](#_Toc190647724)

[3. Program logic and Arithmetic Operators 4](#_Toc190647725)

[4. User inputs and concatenation 4](#_Toc190647726)

[Flowchart (if applicable): 5](#_Toc190647727)

[Python Code: 6](#_Toc190647728)

[Summary 9](#_Toc190647729)

[References 9](#_Toc190647730)

# Assignment 2

|  |
| --- |
| Question #:Use of selections: Write a program that would use nested if /if-else. In this program a variable called temper stores a randomly generated number between 1 and 400. If the generated value is above 100, the program prints “ temperature above boiling point” and uses a second if to check whether the temper is above 320 degrees; in that case the program prints “temperature above smoke point”. For temper lower than 100 the program prints “ temperature is not very high”. Flowchart is required in this program. Student should also suggest what are some of the weaknesses in this problem Python Code: Selections.py  *import random*  *#Generating temperature using random function between 1 to 400*  *temperature = random.randint(1, 400)*  *if temperature > 100:*  *print(f'temperature: {temperature} --temperature above boiling point')*  *if temperature >  320:*  *print(f'temperature: {temperature} --temperature above smoke point')*  *if temperature < 100:*  *print(f'temperature: {temperature} --temperature is not very high')*  Flowchart (if applicable): Paste the screenshot of your output here   Repository (if applicable): State your git repository and give a screenshot of the directory contents (if applicable)    [Semester-2-programming-assignment-2/Selection.py at main · mhmukry/Semester-2-programming-assignment-2](https://github.com/mhmukry/Semester-2-programming-assignment-2/blob/main/Selection.py) Use of for loop: Write a program that uses for loop to calculate the magnitude of a vector, where all dimensions (x,y,z) of the vector are integers entered by the user. The formula for magnitude of vectors is . Flowchart is required in this program. Python Code: Assignment\_2\_for\_loop.py  *import math #Importing math module to use square root function*  *#Initializing the dimension list and vector value*  *dimension\_list = [0,0,0]*  *total\_square\_value\_of\_dimensions = 0*  *#Using for loop to get the user input value to reset dimension list value and calculate the square for the dimension values entered*  *for dimension\_value in dimension\_list:*  *dimension\_value += int(input("Enter the value for the dimension: "))*  *total\_square\_value\_of\_dimensions += dimension\_value\*\*2*  *#print(f'total\_square\_value\_of\_dimensions : {total\_square\_value\_of\_dimensions }')*  *#Finding the square root for |M|= √(x^2+y^2+z^2)*  *vector\_value = math.sqrt(total\_square\_value\_of\_dimensions )*  *print(f'Magnitude of vectors is |M|= {vector\_value}')*  Flowchart (if applicable): Paste the screenshot of your output here   Repository (if applicable): State your git repository and give a screenshot of the directory contents (if applicable)    [Semester-2-programming-assignment-2/Assignment\_2\_for\_loop.py at main · mhmukry/Semester-2-programming-assignment-2](https://github.com/mhmukry/Semester-2-programming-assignment-2/blob/main/Assignment_2_for_loop.py) Use of while loop:  1. Write a program that takes integer inputs from a user in a while loop in a variable called *num* and prints the value of *num*. The loop terminates if the user types a negative number. The program should also count the number of times the loop iterates. Student should provide a flow chart for this problem 2. Use version management with git and make the following changes to the program in part a: The loop breaks if the user types your student number print the message “ cutoff point”. The loop should skip the statements in current iteration and does not increment count whenever the user types a multiple of 11. Flowchart is required for this program  Use of Functions  1. Study the concept of lists , how to initialize a list and how to pass list as an argument to functions 2. Write a program to calculate standard deviation of a list of integers. This program should use a function called *Avg()* that calculates and returns average of the list passed as a parameter. The main program passes that average to another function called *sumSqDiff()* that also accepts a list as an argument and returns the sum square of the difference of list values and the average. sumSqDiff() implements this formula where *xi* is the item in the list at index *i*, and is the average of all items. The main program uses the following formula to calculate and print the standard deviation where σ is the standard deviation, *ssd* is the value returned from *sumSqDiff()*  and N is the total number of elements in the list. Use any loop of your choice. Student should generate a separate flow chart for every function as well as the calling program.  Use of nested if-else, Elif and loops  1. Write a program that takes number of rows  *row* as input from the user. The program counts from 1 up to *row* to generate the following pattern. Note that all odd numbered rows get ‘+‘ symbols whereas even numbered rows get ‘-’ symbol. The program should use the operator \* to generate desired pattern. Student should provide a flowchart for this program   A white background with black text  Description automatically generated Python Code: Assignment\_2A.py  *#Taking user input*  user\_input = int(input("Enter the number of rows: "))  *#Using for loop to print the row for the desired character*  for i in range(1,user\_input+1):  *#Checking if row number is even or odd*      if (i % 2 == 0):  *#Padding space to print the character according to the requested format*          print((user\_input - i)\*" ",(2\*i-1)\*"+")      else:          print((user\_input - i)\*" ",(2\*i-1)\*"\*")  Flowchart (if applicable): Paste the screenshot of your output here   Repository (if applicable): State your git repository and give a screenshot of the directory contents (if applicable)    [**Semester-2-programming-assignment-2/Assignment\_2A.py at main · mhmukry/Semester-2-programming-assignment-2**](https://github.com/mhmukry/Semester-2-programming-assignment-2/blob/main/Assignment_2A.py)   1. Write a program that uses a list called *scores* initialized with 10 values. The program iterates over the list and determines student’s grade for the given score, as done in week3 lesson. Student must use elif construct to determine the grades based on the following criteria: score>80 is grade A, score between 70 and 80 is grade B, score between 60 and 70 is grade C, score between 50 and 60 is grade D and score less than 50 is grade D. Use any loop of your choice. Student must provide a flowchart for this program  Python Code: Assignment\_2B.py  *#Providing a list of scores of each student*  *score\_list = [50, 76, 98, 21, 54, 87, 66, 45, 78, 96]*  *#For loop to go through the entire score list*  *for student\_score in score\_list:*  *#Grade A if score is greater than 80*  *if student\_score >80:*  *print(f'score: {student\_score}  grade A')*  *#Grade B if score is between 70 and 80*  *elif student\_score >= 70 and student\_score <= 80:*  *print(f'score: {student\_score}  grade B')*  *#Grade C if score is between 60 and 69*  *elif student\_score >= 60 and student\_score < 70:*  *print(f'score: {student\_score}  grade C')*  *#Grade D if score is between 50 and 59*  *elif student\_score >= 50 and student\_score < 60:*  *print(f'score: {student\_score}  grade D')*  *#Grade D if less than 50*  *else:*  *print(f'score: {student\_score}  grade D')*    Flowchart (if applicable): Paste the screenshot of your output here   Repository (if applicable): State your git repository and give a screenshot of the directory contents (if applicable)    [Semester-2-programming-assignment-2/Assignment\_2B.py at main · mhmukry/Semester-2-programming-assignment-2](https://github.com/mhmukry/Semester-2-programming-assignment-2/blob/main/Assignment_2B.py) Flowchart (if applicable): Draw a flowchart of your problem and give a title to your figure. For example, your problem is to calculate average sales, then figure title would be “Fig1. Flowchart of average sales”  A diagram of a game  Description automatically generated  A diagram of a work flow  Description automatically generated  Fig #. Flowchart of the MyGame.py Python Code: MyGame.py  import random *#importing random from the general library*  class MyGame():    *#Printing welcome and taking input for each player's name*      print("Welcome to my game")      Player1 = input("Enter the name for Player 1: ")      Player2 = input("Enter the name for Player 2: ")      Player3 = input("Enter the name for Player 3: ")  *#Taking input for first player and printing the first player's value*      input(f'Player1 ({Player1}): Please press enter to roll your dice')      Player1\_dicevalue = random.randint(1, 6)      print(f'Player1 value: {Player1\_dicevalue}')  *#Taking input for second player and printing the second player's value*      input(f'Player2 ({Player2}): Please press enter to roll your dice')      Player2\_dicevalue = random.randint(1, 6)      print(f'Player2 value: {Player2\_dicevalue}')  *#Taking input for third player and printing the third player's value*      input(f'Player3 ({Player3}): Please press enter to roll your dice')      Player3\_dicevalue = random.randint(1, 6)      print(f'Player3 value: {Player3\_dicevalue}')  *#Comparing Player 1's dice value with Player 2's and Player 3's*      if Player1\_dicevalue > Player2\_dicevalue and Player1\_dicevalue > Player3\_dicevalue:          print(f'Player1 ({Player1}) with the value of {Player1\_dicevalue}: wins the game')    *#Comparing Player 2's dice value with Player 1's and Player 3's*      elif Player2\_dicevalue > Player1\_dicevalue and Player2\_dicevalue > Player3\_dicevalue:          print(f'Player2 ({Player2})  with the value of {Player2\_dicevalue}: wins the game')  *#Comparing Player 3's dice value with Player 2's and Player 1's*      elif Player3\_dicevalue > Player1\_dicevalue and Player3\_dicevalue > Player2\_dicevalue:          print(f'Player3 ({Player3})  with the value of {Player3\_dicevalue}: wins the game')  *#Checking if player 1's dice value equals*      elif (Player1\_dicevalue == Player2\_dicevalue) and (Player2\_dicevalue > Player3\_dicevalue):          print(f'Tie between Player 1 ({Player1}) and Player 2 ({Player2}) with the value of {Player1\_dicevalue}: ')      elif (Player1\_dicevalue == Player3\_dicevalue) and (Player3\_dicevalue > Player2\_dicevalue):          print(f'Tie between Player 1 ({Player1}) and Player 3 ({Player3}) with the value of {Player3\_dicevalue}: ')      elif (Player2\_dicevalue == Player3\_dicevalue) and (Player2\_dicevalue > Player1\_dicevalue):          print(f'Tie between Player 2 ({Player2}) and Player 3 ({Player3}) with the value of {Player3\_dicevalue}: ')      elif (Player1\_dicevalue == Player2\_dicevalue) and (Player2\_dicevalue == Player3\_dicevalue):          print(f'Tie between Player 1 ({Player1}), Player 2 ({Player2}) and Player 3 ({Player3}) with the value of {Player3\_dicevalue}: ') |
|  |
|  |

# Summary

I have learned how to:

1. print messages
2. how to take user input
3. how to use python random generator library
4. how to check and execute various program condition logic (if/else if/else)
5. how to use loop to repeat similar task (for loop for printing greeting message multiple times)

# References

1. Downey, A. (2012). Think python. " O'Reilly Media, Inc.".