Addis Ababa University
Addis Ababa Institute of Technology
School of Electrical and Computer Engineering

Assignment

Instructions: You are not allowed to use any data storages like list, dictionaries,sets or tuples

1. Write a program that does the following kind of computation given two numbers
   - For Example given 15 and 20
     - It would produce ->
       - 1 * 20 = 20
       - 3 * 17 = 51
       - 5 * 14 = 70
       - 7 * 11 = 77
       - 9 * 8 = 72
       - 11 * 5 = 55
       - 13 * 2 = 26
     - Done

2. Write a program that would calculate the mean of a sequence of numbers received from a user one at a time until the person inserts the key term "done".

3. Write a program that would take the name, the age and the height of a person and inform the user of seats allowed for them in an airplane.
   3.1.   If the age of a person is less than or equal to 15, they are not allowed to take the window seat
   3.2.   If and only if the age of the person is greater than 20 and their height is greater than 1.65, they are allowed to sit by the exit doors.
   3.3.   If and only if the age of the person is greater than 15 or less than 60, they are allowed to seat in the middle seat
   3.4.   If their age is greater than 60 or less than 15, they should sit by the aisle seat

4. Write a program that would sequentially (not recursive) calculate the following type of sum.
   Eg:- ("^" means to the power of)
   f(3)
   => it prints the following sequence and the computational result
   [3 * (1)^3 + 3 * (2)^3 + 3 * (3)^3] + [2 * (1)^2 + 2 * (2)^2] + [1 * (1)^1] = 119

   f(4)
   => it prints the following sequence and the computational result
   [4 * (1)^4 + 4 * (2)^4 + 4 * (3)^4 + 4 * (4)^4]+ [3 * (1)^3 + 3 * (2)^3 + 3 * (3)^3] + [2 * (1)^2 + 2 * (2)^2] + [1 * (1)^1] = 1535

5. Write a program that asks the name of a student and then it asks for the father's name of the student if you know it then the father name of the father's name and so forth like this:-

> What is the name of the student? abebe
> Do you know the father's name of 'abebe' ? y/n y
> What is the name of the father of 'abebe' ? kebede
> Do you know the father's name of 'kebede' ? y/n y
> What is the name of the father of 'kebede' ? alemu
> Do you know the father's name of 'alemu' ? y/n h
> Do you know the father's name of 'alemu' ? y/n j
> Do you know the father's name of 'alemu' ? y/n n
>
> The student's full name is * abebe kebede alemu *
>
> any more students ? y/n y
> what is the name of the student? alemitu
> Do you know the father's name of 'alemitu' ? y/n y
> what is the name of the father of 'alemitu' ? abebe
> Do you know the father's name of 'abebe' ? y/n n
>
> The student's full name is * alemitu abebe *
>
> any more students ? y/n n

The words in blue are user inputs and those in balck are output of the program.
The question "Do you know the father's name of —--?" would be displayed if the user inputs anything different from y/n
The question "any more students ?" would exit if any response other than 'y' is given

6. Write a program that would take a number from a user and prints out the multiplication table of the number upto a random number between an interval of your choosing. and then check if the multiple is a perfect square (is a square of another number or has an integer square root) and print out "it is perfect square" next to it or if it is a 3rd root of any number, it prints " it is a perfect 3rd root". It would give the user a random amount of chances for each run to provide a number. Example output ->

> you have 2 chances
> Give me a number - 6
> 6 * 1 = 6
> 6 * 2 = 12
> 6 * 3 = 18
> 6 * 4 = 24
> 6 * 5 = 30
> 6 * 6 = 36 => it is a perfect square of 6
> 6 * 7 = 42
> 6 * 8 = 48

6 * 9 = 54
Give me a number - 4
4 * 1 = 4 => it is a perfect square of 2
4 * 2 = 8 => it is a perfect third root 2
4 * 3 = 12
4 * 4 = 16 => it is a perfect square of 4
4 * 5 = 20
4 * 6 = 24

7. Write a program given a global string variable, it would allow the user to guess the spelling of the word one letter at a time (implement the hung man game). You are not allowed to use lists or dictionaries or tuples or any other data structure. To simplify, use a word with non repeated letters. The number of tries for guessing is the length of the chosen word plus one. If the same word is provided by the user, the number of tries are not decreased.

Hint :
-> you can use a number to store found and not found letters
Eg. for instance, 1 could represent letters not found and 2 could represent words that are found. So for the word "great" when no letter is found, the tracking number will be 11111 and when all the letters are found then the tracker number would be 22222
-> to check if a letter is in a word use the keyword "in"
-> to find the index of a letter in a word, you can use "find" function like this a="abcd"
b=a.find("c") which returns b=2. An Example output:-

you are left with 6 trys
 Give me a letter - a
  11121   ___a_
you are left with 5 trys
 Give me a letter - a
  11121   ___a_
you are left with 5 trys
 Give me a letter - e
  11221   __ea_
you are left with 4 trys
 Give me a letter - g
  21221   g_ea_
you are left with 3 trys
 Give me a letter - t
  21222   g_eat
you are left with 2 trys
 Give me a letter - i
  ** Nope **
you are left with 1 trys
 Give me a letter - r
  22222   great
you did it!!

8.   Write a program that accepts a string from a user and checks if the string provided is symmetrical or not. Do this until the user types done.
Please enter a string: abeba
String is symmetrical
Please enter a string: abebe
String is not symmetrical
Please enter a string: done
Exit

9.   Write a Python program that accepts an integer (n) and computes the value of n+nn+nnn.
Sample value of n is 5
Expected Result : 615

10.   Write a program that accepts n (a number) from a user and generate n-digit password with only numbers.
Input from user: 6
Output: 589452

11.   Write a program that accepts a number from a user between 1 and 10 and prints the following triangular format.
Input from user: 4
output:
1
22
333
4444

12.   Write a program that accepts a number from the user between 1 and 10 and prints n number of lines of the following pattern.
Input from user:7
output:
```
      *
     * *
    * * *
   * * * *
  * * * * *
 * * * * * *
* * * * * * *
```

13.   Write a program that sequentially (not recursively) computes the first 10 summation series produced by adding the previous summations of "setSize" given by the user. When setSize=2 it becomes a fibonacci series. For example for sets of size 2, 3 and 4 ;

   ● For sets of size two (Fibonacci) => f(n)=f(n-1)+f(n-2)

- for sets of size three => f(n)=f(n-1)+f(n-2)+f(n-3)
- for sets of size four => f(n)=f(n-1)+f(n-2)+f(n-3)+f(n-4)

The outputs would look like the following (you do not need to include the addition part)

Give me set size- 3

- f(0)= 0
- f(1)= 1
- f(2)= 2
- f(3)= 0+1+2 = 3
- f(4)= 1+2+3 = 6
- f(5)= 2+3+6 = 11
- f(6)= 3+6+11 = 20
- f(7)= 6+11+20 = 37
- f(8)= 11+20+37 = 68
- f(9)= 20+37+68 = 125

- Since you are not allowed to use lists, you can use a string to hold the previous summed values.

- TRY to come up with an algorithm that would compute the above on your own. If you have tried it all but it did not work, then follow the following steps

13.1. Write a function that would take a string of numbers separated by a delimiter of your choice and produce the sum of those numbers. Eg. given a="11_12_3_4_", it returns 30.

➔ Hint

◆ using the "myString.find(sub_string)" function on a string 'myString' returns the first index of the sub_string. Eg. b= a.find("_") gives b= 2 .

◆ So to find the next occurrence of the sub_string, you can start from the previous index plus one. E.g. b= a.find("_",3) gives b= 5

◆ If the sub_string is not found it returns -1. E.g. b=a.find("_",10)

◆ Substrings can be formed using myString[start:end]. E.g. c=a[3,5] gives c="12"

13.2.   Write a function that produces the nth value [f(n)] for the given set size.

➔ If n is less than set size it will return the number itself

➔ Else it will add the sequence numbers of set size before it. It stores the previous sums in a string and uses the function created in 13.1 to get the next sum.

◆ E.g. in case of set size 3 and n equal to 6 the loop would go through

● fun_13.1("0_1_2_")= 3

● fun_13.1("1_2_3_")= 6

● fun_13.1("2_3_6_")= 11

● fun_13.1("3_6_11_")= 20

13.3.   Write a function that would call the function in 13.2 in an iteration until sequence size (10 in this case but you can take this from the user as well) and pass the user selected set size.