COAL CS FALL 2020
BSCSF20
Assignment # 1

You are required to prepare the ASM file of each task. Submit All ASM files. Please create a RAR file with your roll number containing all tasks.

Deadline: Thursday, 23 December 2021, 11:00 AM

Task 1

Write a program that will read a string from the user (**assume that string only contain capital and small letters without spaces end by a carriage return**), After input you have to display the given string in ascending order. ***Note: All capital letters are smaller than the small letters. Pseudo code for sorting is given below.***

(Input is underlined in sample run, only to distinguish from display messages)

Sample Run 1:

```
Enter the String:AXrLzCoyBZD


Sorted String is:ABCDLXZoryz
```

Sample Run 2:

```
Enter the String:deMAiZWmb


Sorted String is:AMZWbdeim
```

Hint:

Pseudo code for sort any array in ascending order:

```
int arr [10]= {10, 3, 1, 4, 9, 34, 41, 2, 22, 5 }
for (int i=0 ; i < 10; i++)
{
    for (int j= i+1; j < 10; j++)
    {
        if( a[i] > a[j])
        {
            int temp = a[i];
            a[i] = a[j] ;
            a[j] = temp;
        }
    }
}
```

Task 2

Write an Assembly Language Program that lets the user type in an algebraic expression, ending with a carriage return that contains parenthesis only. As the expression is being typed in, the program evaluates each character. If at any point the expression is incorrectly bracketed [too many left or right], the program tells the user to start over. After the carriage return is typed, if the expression is correct, the program displays "expression is correct" and the program asks the user if he or she wants to continue. If the user types 'Y', the program runs again. If the expression is not correct, the program displays "too many left brackets, begin again" or “too many right brackets, begin again” according to expression. Your program does not need to store the input string, only check it for correctness. **[Use stack for this task]**

Sample execution:

Enter an algebraic expression: **a+b)**

Too many right brackets, Begin again

Enter an algebraic expression: **a+(b-c**

Too many left brackets, Begin again

Enter an algebraic expression: **a+(b-c)**

“Expression is correct”

Type Y if you want to continue: **N**

Task 3

Being a good student hope you will create new CPU design as grand assignment of COAL subject. Suppose whole class copies it from one source. And that source that thinks is smart does not create all circuit of CU (Control Unit) (for example source skips add, multiply and power). He provides only one function and that is increment. After submission instructor becomes aware of this Democratic Cheating. In thunder he states that he will give all of you “F” grade (Sound Familiar). As usual you said sorry because it’s the only asset you always have in this situation. Instructor assigns a penalty by saying “You have not provided add, multiply and power functions using design. Now your penalty is to provide these functions using programing.”

Sound familiar story. Now your task is to provide following procedures:

- **Adder (uses increment statement as you know 5+4 = 5+1+1+1+1 (4 times increment))**
- **Multiply (uses increment statement as you know 5*4 = 5+5+5+5 (4 times add 5))**
- **Power ((uses increment statement as you know 5^4 = 5*5*5*5 (4 times multiply 5)))**
- **Main Procedure which provides all above task functionality as calculator.**