COS 135 Individual Assignment Week 6

Due: Friday 03/08/19 End of the day (late submissions -10%)

This assignment has 2 sections (part #1 and part #2) in **5 pages**. Please submit a .zip file with answers (**answering template is attached for part #1**) and complete source code/s for part #2.

Part #1 (24pts) Select or write the most appropriate answer (please use the answering template provided).

1. what is the output of following program?

```
#include<stdio.h>

int main()
{
    printf("%d",printf("COS135"));
}
```

a) COS135

b) COS1356

c) 6COS135

d) 6

2. What will output when you compile and run the above code?

```
#include<stdio.h>

int main()
{
    int a=5,b=6,c=11;
    printf("%d %d %d");
}
```

a) Garbage value garbage value garbage value

b) 5 6 11

c) 11 6 5

d) Compiler error

3. What will be output of following c code?

```c
#include<stdio.h>

int main(){
    int x = 011;
    for(int i = 0; i < x; i += 3)
    {
        printf("Start");
        continue;
        printf("End");
    }
    return 0;
}
```

a) StartStartStart

b) EndEndEnd

c) StartEndStartEndStartEnd

d) StartStartStartStart

4. What will be output of following c code?

```c
#include<stdio.h>

int main()
{
    static int i;
    for(++i; ++i; ++i)
    {
        printf("%d", i);
        if(i == 4) break;
    }
    return 0;
}
```

a) 24

b) Program terminate without output

c) Compile error

d) 2

5. How many times this do.. while.. loop will execute?

```
#include<stdio.h>
int main()
{
    char c = 125;
    do{
        printf("%d ", c);
    } while(c++);
    return 0;
}
```

e) 3

f) 132

g) Infinite loop

h) Never execute the loop

6. What is the output of following program?

```
#include<stdio.h>
int main()
{
    int a = 320;
    char *ptr;
    ptr =( char *) &a;
    printf("%d ", *ptr);
    return 0;
}
```

a) 2

b) 320

c) 64

d) Compilation error

Part #2 (76pts): write C programs for following tasks and submit your source code/s

Special instructions:

Comments are required in the following locations:

- At the top of the source code comment your name and a short program description.
- Comment the purpose of variables, functions, and other elements in your code.
- Comment major sections of code such as input, processing, and output.

Program Design:
Your program is a professional document and must be neat and easy to read. All programs should follow the listed specifications.

- Comments should be aligned and entered in a consistent fashion
- Blank lines should be added to aid readability
- Code within blocks should be indented
- Comments should not contain spelling mistakes
- Variable names should be meaningful
- Define functions and data structures where necessary
- Optimize your code: least possible number of lines to produce the output
- Error handling: you should handle all the possible error conditions and invalid inputs

(a) **(16pts)**: The Fibonacci sequence is a series where the next term is the sum of pervious two terms. The first two terms of the Fibonacci sequence is 0 followed by 1.

The Fibonacci sequence: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144…….. N

Write a program to output the Fibonacci sequence up to a given number by a user. Your program should verify the user's input as a positive integer.

Sample output #1:

Enter a number: 90

Fibonacci Series: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89

Sample output #2:

Enter a number: -9

Invalid input – please try a positive integer.

(b) **(60pts)**: Write a C program to validate a user's password. A user should be able to enter his/her preferred password via keyboard input. Your program checks for specific criteria given below and returns **Valid Password** or **Invalid Password**. If the user's preferred password is invalid, it also outputs different reasons.

Write separate functions to check the following criteria of the preferred password (each function takes user entered password as a parameter, and returns **1** if it meets a certain criteria else returns **0**):

- A function to test the password is at least eight characters long
- A function to test the password has at least one uppercase letter and one lower case letter
- A function to test the password has at least one digit
- A function to test the password has at least one of these four characters: ! @ # $

Sample output #1:

```
Please enter your preferred password: UOM@orono2018
Valid
```

Sample output #2:

```
Please enter your preferred password: UoM2018
Invalid
Reason1: your password should contain at least eight characters.
Reason2: your password should contain at least one of the four special characters "! @ # $"
```

Now, extend your code to encrypt a valid password. Once your program detects a valid password, apply below **encryption algorithm** and output the encrypted password. You may define new functions where necessary.

Encryption algorithm is based on corresponding ASCII values of different characters in the valid password. For each character in the password,

- add +1 if it is alphanumeric. For example, if the character is a uppercase letter, lower case letter, or a digit: encrypted_character = (character + 1);

- add -1 if it is one of the four special characters. For example, if the character is one of these four characters ! @ # $: encrypted_character = (character - 1);

Example:

UOM@orono2018 should be encrypted as **VPN?pspop3129**