Green University of Bangladesh
Department of Computer Science and Engineering
CT3 (Assignment), Summer 2022
Course Code: CSE 103 Course Title: Structured Programming

1. Write a C program that takes a string as input and reverses each of the word without using built in function.

Solution of 01:

1. String reverse wordByWord.c > ...

```c
#include<stdio.h>
void  stringReverseWordByWord(char str[], int start, int end)
{
    char temp;
    while(start<end)
    {
        temp=str[start];
        str[start]=str[end];
        str[end]=temp;
        start++;
        end--;
    }
}
int main()
{
    char str[1000];
    int start = 0, end = 0;
    printf("Enter a string to reverse word by word: ");
    fgets(str, sizeof(str), stdin);
    //printf("%d",str[end] );
    while(str[end]){
        for (end = start;str[end]&&str[end]!=' '; end++);
        stringReverseWordByWord(str, start, end-1);
        start=end+1;
    }
        puts(str);
    return 0;
}

```

Green University of Bangladesh
Department of Computer Science and Engineering
CT3 (Assignment), Summer 2022
Course Code: CSE 103 Course Title: Structured Programming

Output of question 01:

```
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ gcc 1.\ String\ reverse\ wordByWord.c -o "1. String Reverse"
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ ./'1. String Reverse'
Enter a string to reverse word by word: abc xyz
cba
zyx
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ abc 123
Command 'abc' not found, but there are 17 similar ones.
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ ./'1. String Reverse'
Enter a string to reverse word by word: abc 123
cba
321
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ ./'1. String Reverse'
Enter a string to reverse word by word: abx @#1
xba
1#@
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ -
```

2. Write a C program that takes a string as input and finds the length of that string using recursive function.

Solution of Question 2:

2. srtlen using recursive.c > ...

```c
// String Length using Recursion
// Created by: Jahidul Islam
// Date: 2022-09-09
#include<stdio.h>
#define SIZE 10000

int stringLenth(char str[], int start)
{
    if(str[start]=='\0')
        return start;
    else
        return stringLenth(str, start+1);
}
int main(){
    char str[SIZE];
    int start = 1;
    printf("Enter a string to find the length: ");
    fgets(str, sizeof(str), stdin);
    //printf("The length of the string is: %d\n", stringLenth(str, start));
    int size = stringLenth(str, start);
    printf("The length of the string is: %d\n", size-1); // -1 for last index which is '\0'
    return 0;
}
```

Green University of Bangladesh
Department of Computer Science and Engineering
CT3 (Assignment), Summer 2022
Course Code: CSE 103 Course Title: Structured Programming

Output of question 2:

3. Suppose you have some words and you want to right justify them, that is, align them to the right. Create a program that reads some words and print it all right justified, in the same order as they appear in the input.

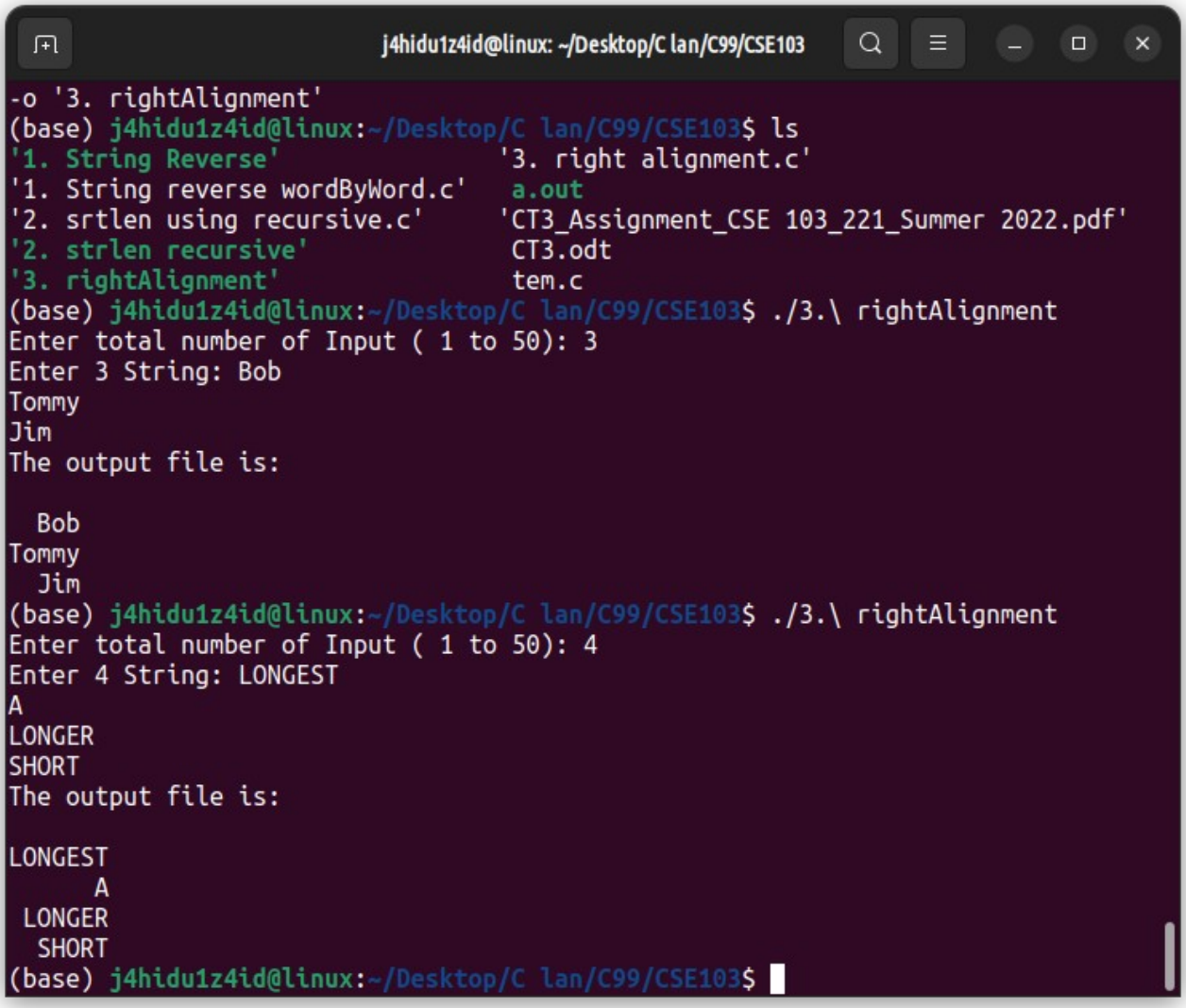
Solution of question 03:

3. right alignment.c > main()

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main(){
    printf("Enter total number of Input ( 1 to 50): ");
    int n;
    scanf("%d", &n);
    if(n<1 || n>50){
        printf("Invalid Input");
        exit(0);
    }
        printf("Enter %d String: ", n);
        //declare 50 char arrays
        char str[50][100];
        for (int i = 0 ; i < n; i++)
        {
            scanf("%s", str[i]);
        }
        //find the longest string
        int max = 0;
        for (int i = 0; i < n; i++)
        {
            if(strlen(str[i])>max)
                max = strlen(str[i]);
        }
        //printf("%d", max);
        printf("The output file is: \n\n");
        for (int i = 0; i < n; i++)
        {
            printf("%*s", max, str[i]);
            printf("\n");
        }
        return 0;
}
```

Green University of Bangladesh
Department of Computer Science and Engineering
CT3 (Assignment), Summer 2022
Course Code: CSE 103 Course Title: Structured Programming

Output of question 03:



Green University of Bangladesh
Department of Computer Science and Engineering
CT3 (Assignment), Summer 2022
Course Code: CSE 103 Course Title: Structured Programming

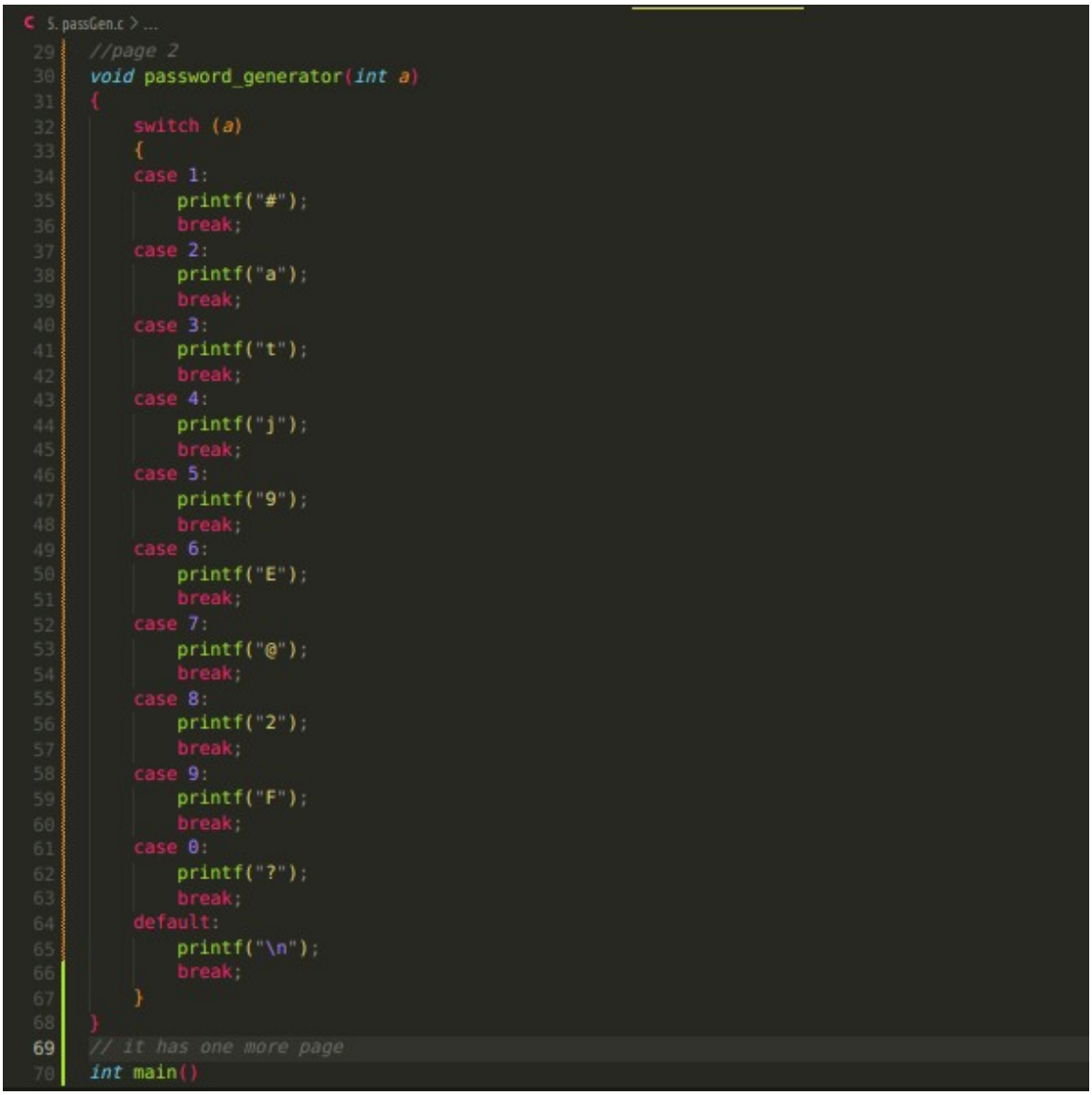
4. Cyber Security is a key issue to protect our daily documents and applications stored and submitted in various platforms. Having a robust encryption system to our generated password is very essential in this perspective. Your task is to create a nice and smooth encrypted password generator. Follow the instructions carefully to build the password generator. [5 Marks]

Solution of question 04:

5. passGen.c > ...

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int digitCount(int n)
{
    if (n == 0)
    {
        return 0;
    }
    return 1 + digitCount(n / 10);
}
int Input_Number()
{
    int n;
    printf("Enter a 5 digits number: ");
    scanf("%d", &n);
    if (digitCount(n) != 5)
    {
        printf("Wrong Input\n");
        exit(0);
    }
    return n;
}
// it has one more page
```

Green University of Bangladesh
Department of Computer Science and Engineering
CT3 (Assignment), Summer 2022
Course Code: CSE 103 Course Title: Structured Programming



```c
// it has one more page
int main()
{
    int recived, temp;
    int i, j, k;
    int str[5];

    recived = Input_Number();

    for (i = 0; i <= 5; i++)
    {
        temp = recived;
        recived = temp % 10;
        for (j = 5; j > i; j--)
        {
            str[j] = recived;
        }
        temp = temp / 10;
        recived = temp;
    }
    // reverse the array and pass it to the password generator
    for (k = 5; k >= 0; k--)
        {
            password_generator(str[k]);
        }
    printf("\n\n");
}

//end of the code
```

Output of question 04:

Green University of Bangladesh
Department of Computer Science and Engineering
CT3 (Assignment), Summer 2022
Course Code: CSE 103 Course Title: Structured Programming

5. In programming terms a recursive function can be defined as a routine that calls itself directly or indirectly. Using recursive algorithm, certain problems can be solved quite easily.
Write a program using recursive function that can convert a Decimal number to its equivalent Binary number.

Solution of question 05:

4. decimal to binary using recursion.c > ...

```c
#include<stdio.h>
int decToBinary(int n)
{
    if (n == 0)
        return 0;
    else
        return (n % 2 + 10 * decToBinary(n / 2)); //recursive call
}
int main()
{
    int n;
    printf("Enter a decimal number: ");
    scanf("%d", &n);
    // print and calling the recursive function
    printf("%d\n", decToBinary(n));
    return 0;
}
```

Output of question 05:

```
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ gcc 4.\ decimal\ to\ binary\ using\ recursion.c -o '4. decToBin recF'
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ ./4.\ decToBin\ recF
Enter a decimal number: 3
11
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ ./4.\ decToBin\ recF
Enter a decimal number: 5
101
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$ ./4.\ decToBin\ recF
Enter a decimal number: 11
1011
(base) j4hidu1z4id@linux:~/Desktop/C lan/C99/CSE103$
```

6. Write a program that will correctly decode a set of characters into a valid message. Your program should read a given file of a simple coded set of

characters and print the exact message that the characters contain. The code key for this simple coding is a one for one character substitution based upon a single arithmetic manipulation of the printable portion of the ASCII character set.

Solution of question 06:

output 06:

1.

Green University of Bangladesh
Department of Computer Science and Engineering
CT3 (Assignment), Summer 2022
Course Code: CSE 103 Course Title: Structured Programming

2.

3.