ULAB

UNIVERSITY OF LIBERAL ARTS BANGLADESH

ASSIGNMENT-2

Submitted To:

Niaz Ashraf Khan
Lecturer
University of Liberal Arts Bangladesh

Submitted By:

Name: Md Abul Hashem
Roll: 222014097
Course : CSE1201
Section: 2

Problem on String 1

Write a program in C to check whether a given string is palindrome or not.

Problem on String 2

Write a program in C to reverse order of words in a string.

For example:

Input string : I love ULAB

ExpectedOutput: ULAB love I

Input string : Structured Programming Theory

ExpectedOutput: Theory Programming Structure

Problem on Structure 1

1. Write a C/C++ Program using struct variables to construct a student record list consisting of the following information in each record: student id (integer), student name (character string) and Marks (integer).

Do the following operations:

a. Store data for ten students and display the data on the screen.

b. Consider that students Marks and find out student names who get maximum and minimum marks

c. Search a record based on student name and display the information content.

If the specified name is not present in the list an error message should be displayed.

[Problem on String 1]

```
#include <stdio.h>
#include<string.h>

int main()
{
    int i,j;
    char arra[200];
    fgets(arra, sizeof(arra),stdin);
    int len;
    len = strlen(arra)-1;
    for(i=0; i<=len; i++)
    {
       if(arra[i]!= arra[len-i-1])
       {
          printf("This is Not Palindrome");
          break;
       }
       else
       {
          printf("This is palindrome");
          break;
       }
    }
    return 0;
}
```

[Problem on String -2]

```
#include <stdio.h>
#include<string.h>

int main()
{
   char sen[100];
   char rsen[100];
   int len, i, index;
   int wstart;
   int wend;
   printf("Enter Your String: ");
   gets(sen);

   len = strlen(sen);
   index=0;
   // checking of words from the end
      of string

  wstart = len-1;
  wend =len-1;

  While(wstart >0)
  {
     if(sen[wstart]== ' ')
     {
        // Add to the reverse string
        i=wstart+1;
```

```
            while(i<= wend)
            {
                rsen[index]= sen[i];
                i++;
                index++;
            }
            rsen[index++] = ' ';
            wend = wstart -1;
        }
        wstart--;
    }
// Add the las word
for(i=0; i<=wend; i++)
{
    rsen [index]= sen[i];
    index++;
}
// Add NULL charcacheter at the end of
 reverse string
rsen [index] = sen[i]'\0';
    printf("\n Reverse Your String: %s"; rsen);
    printf("\n");
    return 0;
```

[Problem on Structure-1]

```
#inleude<stdio.h>
#include<string.h>

struct student
{
    char Name[20];
    int ID;
    int Marks;
};

int main()
{
    int n;
    printf("How Many students List You
            want to Record: ");
    scanf("%d", &n);

    int i;
    struct student S[n];
    int max = S[0].Marks;
    char mxname;
    char miname;
    printf("\n");
    for(i=0;i<=n;i++)
    {
    printf("Enter Name-%d:",i);
    scanf("%s", &S[i].Name);
```

```
        printf("Enter ID-%d", i);
        scanf("%d", &S[i].ID);
        printf("Enter Marks -%d:",i);
        scanf("%d", &S[i].Marks);
        printf("\n");
    }
    for(i=0;i<=n;i++)
    {
        printf("Student Name-%d: %s\n",i,
                S[i].Name);
        printf("Student ID-%d: %d\n", i,S[i].ID);
        printf("Student Marks-%d: %d\n",i,S[i].Marks);
    }
    for(i=0; i<=n; i++)
    {
        if(S[i].Marks > max)
        {
            max = S[i].Marks;
            mxname = i;
        }
    }
    int min = S[0].Marks;
    int t;
    for(t=0;t<=n; t++)
    {
        if(S[t].Marks < min)
```

```
        {
            min = S[t].Marks;
            miname = t;
        }
    }
    int x, found;
    char SName[50];
    scanf("%s", &SName);

    for(x=0; x<=n; x++)
    {
        if(strcmp(S[x].Name, SNam) == 0)
        {
            printf("%s\n", S[x].Name);
            printf("%d\n", S[x].ID);
            printf("%d\n", S[x].Marks);
            break;
        }

        else
        {
            printf("Bhagen mile nai kisu");

        }
    }
    printf("Max name is: %s\n", S[mxname].Name);
    printf("Min name is: %s\n", S[miname].Name);

    return 0;
}
```