

CS-111

<Module No- 7>

NITK SURATHKAL



INBASEKARAN.P

201EC226

Mentor: Mrs Marwa Mohiddin

```

// Lab 7 Questin 1
// Inbasekaran.P 201EC226
/*Program to count frequency of every character present in a line
of text*/

// For printf() and scanf()
#include <stdio.h>
// Including stdlib for system("clear") to clear the screen in the
terminal.
#include <stdlib.h>
#include <string.h>
// Size of the string
#define SIZE 100
#define COUNT_SIZE 128

int *countChar(char *string)
{
    int *count = (int *)malloc((COUNT_SIZE) * sizeof(int));
    if (count == NULL)
    {
        printf("Error!!! memory not allocated.");
        exit(0);
    }
    memset(count, 0, COUNT_SIZE * sizeof(int));
    while (*string)
    {
        count[*string]++;
        string++;
    }
    return count;
}

void displayCount(int *count)
{
    printf("Char : Count\n");
    for (size_t i = 0; i < COUNT_SIZE; i++)
    {
        if(*(count + i))
            printf("%c : %d\n", i, *(count + i));
    }
}

```

```

void inputStr(char *string)
{
    printf("Enter the string: ");
    fflush(stdin);
    gets(string);
    fflush(stdin);
}

int main()
{
    // To clear the console.
    system("clear");
    char string[SIZE] = {'\0'};
    // input the string
    inputStr(string);
    int *count = countChar(string);
    displayCount(count);
    free(count);
    return 0;
}

```

OUTPUT

```
Enter the string: This is a test
```

```
Char : Count
```

```
  : 3
```

```
T : 1
```

```
a : 1
```

```
e : 1
```

```
h : 1
```

```
i : 2
```

```
s : 3
```

```
t : 2
```

```
PS D:\Documents\NIT-K\My_Second_Sem\CS111\Code\M7> |
```

```

// Lab 7 Questin 2
// Inbasekaran.P 201EC226
/*Program to swap two numbers*/

// For printf() and scanf()
#include <stdio.h>
// Including stdlib for system("clear") to clear the screen in the terminal.
#include <stdlib.h>

void swap(int *a, int *b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
    return;
}

int input()
{
    int n;
    printf("Enter the number: ");
    scanf("%d", &n);
    return n;
}

void dispaly(int a)
{
    printf("The number is: %d\n", a);
}

int main()
{
    // To clear the console.
    system("clear");
    int a = input();
    int b = input();
    dispaly(a);
    dispaly(b);
    swap(&a, &b);
    dispaly(a);
    dispaly(b);
    return 0;
}

```

OUTPUT

```

Enter the number: 1
Enter the number: 2
The number is: 1
The number is: 2
The number is: 2
The number is: 1
PS D:\Documents\NIT-K\My_Second_Sem\CS111\Code\M7> |

```

```

// Lab 7 Questin 3
// Inbasekaran.P 201EC226
/*Program to find area and circumference of a circle*/
// For printf() and scanf()
#include <stdio.h>
// Including stdlib for system("clear") to clear the screen in the terminal.
#include <stdlib.h>
// Size of the string
#define PI 3.141

int input()
{
    int n;
    printf("Enter the radius: ");
    scanf("%d", &n);
    return n;
}

double area(int *radius)
{
    return PI * (*radius) * (*radius);
}

double circumference(int *radius)
{
    return 2 * PI * (*radius);
}

int main()
{
    // To clear the console.
    system("clear");
    int r = input();
    printf("The area is: %lf\n", area(&r));
    printf("The circumference is: %lf\n", circumference(&r));
    return 0;
}

```

OUTPUT

```

Enter the radius: 1
The area is: 3.141000
The circumference is: 6.282000
PS D:\Documents\NIT-K\My_Second_Sem\CS111\Code\M7>

```

```

// Lab 6 Questin 4
// Inbasekaran.P 201EC226
/*
Write functions for the following string operations.
a) Concatenation. b) Comparison. c) Length d) Copy e) Reverse.
*/
// For printf() and scanf()
#include <stdio.h>
// Including stdlib for system("clear") to clear the screen in the terminal.
#include <stdlib.h>
#include <string.h>
// Size of the string
#define SIZE 100

int length(char *string);
char *concatenate(char *string1, char *string2);
int comparison(char *string1, char *string2);
char *copy(char *string);
char *reverse(char *string);
void swap(char *a, char *b);
void switch_case(int choice);
int menu();
void inputStr(char *string);
void display(char *string);

int main()
{
    // To clear the console.
    system("clear");
    switch_case(menu());
    return 0;
}

int menu()
{
    printf("Enter Your choice\n");
    printf("1. Concatenation\n");
    printf("2. Comparison\n");
    printf("3. Length\n");
    printf("4. Copy\n");
    printf("5. Reverse\n");
    int choice;
    scanf("%d", &choice);
    return choice;
}

```

```

void display(char *string)
{
    int i = 0;
    while (string[i])
    {
        printf("%c", string[i++]);
    }
    printf("\n");
}

void inputStr(char *string)
{
    printf("Enter the string: ");
    fflush(stdin);
    gets(string);
    fflush(stdin);
}

void switch_case(int choice)
{
    switch (choice)
    {
        case 1:
        {
            char string1[SIZE] = {'\0'};
            // input the string
            inputStr(string1);
            char string2[SIZE] = {'\0'};
            // input the string
            inputStr(string2);
            char *strCat = concatenate(string1, string2);
            display(strCat);
            break;
        }
        case 2:
        {
            char string1[SIZE] = {'\0'};
            // input the string
            inputStr(string1);
            char string2[SIZE] = {'\0'};
            // input the string
            inputStr(string2);
            int n = comparison(string1, string2);
            if (n == 1)
                printf("%s is larger\n", string1);
            else if (n == -1)
                printf("%s is larger\n", string2);
            else
                printf("Same Strings\n");
            break;
        }
    }
}

```

```

case 3:
{
    char string1[SIZE] = {'\0'};
    // input the string
    inputStr(string1);
    int len = length(string1);
    printf("The length of the string is %d", len);
    break;
}
case 4:
{
    char string1[SIZE] = {'\0'};
    // input the string
    inputStr(string1);
    char *string2 = copy(string1);
    printf("The new copy of the string:");
    display(string2);
    break;
}

case 5:
{
    char string1[SIZE] = {'\0'};
    // input the string
    inputStr(string1);
    char *string2 = reverse(string1);
    printf("The reverse of the string:");
    display(string2);
    break;
}
default:
    printf("wrong Input\n");
}
}

int length(char *string)
{
    int n = 0;
    while (string[++n])
        ;
    return n;
}

```



```

char *concatenate(char *string1, char *string2)
{
    int n1 = length(string1);
    int n2 = length(string2);
    int n = n1 + n2;
    char *string = (char *)malloc(n * sizeof(char) + 1);
    if (string == NULL)
    {
        printf("Error!!! memory not allocated.");
        exit(0);
    }
    memset(string, 0, n * sizeof(char) + 1);
    int i = 0;
    for (; i < n1; ++i)
    {
        string[i] = string1[i];
    }
    for (int j = 0; j < n2; ++i, ++j)
    {
        string[i] = string2[j];
    }
    return string;
}

int comparison(char *string1, char *string2)
{
    int i = 0;
    while (string1[i++] && string2[i++])
        ;
    if (string1[i])
        return 1;
    else if (string2[i])
        return -1;
    else
        return 0;
}

char *copy(char *string)
{
    int n = length(string);
    char *stringC = (char *)malloc(n * sizeof(char) + 1);
    if (string == NULL)
    {
        printf("Error!!! memory not allocated.");
        exit(0);
    }
    memset(stringC, 0, n * sizeof(char) + 1);
    for (size_t i = 0; i < n; i++)
    {
        stringC[i] = string[i];
    }
    return stringC;
}

```

```

void swap(char *a, char *b)
{
    char temp = *a;
    *a = *b;
    *b = temp;
    return;
}

char *reverse(char *string)
{
    int i = 0;
    int j = length(string) - 1;
    while (j > i)
    {
        swap(&string[i++], &string[j--]);
    }
    return string;
}

```

OUTPUT

Enter Your choice

1. Concatenation

2. Comparison

3. Length

4. Copy

5. Reverse

1

Enter the string: CS

Enter the string: 111

CS 111

2

Enter the string: cs

Enter the string: cs

Same Strings

3

Enter the string: inba

The length of the string is 4

4

Enter the string: test

The new copy of the string:test

```

// Lab 5 Questin 5
// Inbasekaran.P 201EC226
/*Write a program to display the greatest of N numbers -use malloc() function*/

// For printf() and scanf()
#include <stdio.h>
// Including stdlib for system("clear") to clear the screen in the terminal.
#include <stdlib.h>
#include<limits.h>

int main()
{
    // To clear the console.
    system("clear");
    int n;
    printf("Enter the total number of elements: ");
    scanf("%d", &n);
    int *data = (int *)malloc(n * sizeof(int));
    if (data == NULL)
    {
        printf("Error!!! memory not allocated.");
        exit(0);
    }
    for (int i = 0; i < n; ++i)
    {
        printf("Enter number%d: ", i + 1);
        scanf("%d", (data + i));
    }
    int max = INT_MIN;
    for (int i = 1; i < n; ++i)
    {
        if (max < *(data + i))
        {
            max = *(data + i);
        }
    }
    printf("Largest number is= %d", max);
    free(data);
    return 0;
}

```

OUTPUT

```

Enter the total number of elements: 4
Enter number1: 1
Enter number2: 2
Enter number3: 4
Enter number4: 8
Largest number is= 8

```

```

// Lab 7 Questin 6
// Inbasekaran.P 201EC226
/*Write a program to arrange N names in alphabetical order using
dynamic memory allocation*/

// For printf() and scanf()
#include <stdio.h>
// Including stdlib for system("clear") to clear the screen in the terminal.
#include <stdlib.h>
#include <string.h>
// Size of the string
#define SIZE 100

char *inputName()
{
    printf("Enter the number of characters in the name: ");
    int n = 0;
    scanf("%d", &n);
    char *string = (char *)malloc(n * sizeof(char) + 1);
    if (string == NULL)
    {
        printf("Error!!! memory not allocated.");
        exit(0);
    }
    memset(string, 0, n * sizeof(char) + 1);
    fflush(stdin);
    printf("Enter the name: ");
    for (size_t i = 0; i < n; i++)
    {
        scanf("%c", &string[i]);
    }
    fflush(stdin);
    return string;
}

```

PTO

```

char **nameList()
{
    printf("Enter the number of students: ");
    int n = 0;
    scanf("%d", &n);
    char **list = (char **)malloc(n * sizeof(char *) + 1);
    if (*list == NULL)
    {
        printf("Error!!! memory not allocated.");
        exit(0);
    }
    for (size_t i = 0; i < n; i++)
    {
        list[i] = inputName();
    }
    list[n] = NULL;
    return list;
}

void displayList(char **list)
{
    printf("Name list: \n");
    int i = 0;
    while (list[i])
    {
        printf("%s\n", list[i++]);
    }
}

void swap(char **name1Add, char **name2Add)
{
    char *temp = *name1Add;
    *name1Add = *name2Add;
    *name2Add = temp;
    return;
}

```

PTO

```

int len(char **list)
{
    int i = 0;
    while (list[i])
    {
        i++;
    }
    return i;
}

void sortList(char **list)
{
    int n = len(list);
    for (int i = 0; i < n - 1; i++)
    {
        for (int j = 0; j < n - i - 1; j++)
        {
            if (list[j][0] > list[j + 1][0])
            {
                swap(&list[j], &list[j + 1]);
            }
            else if (list[j][0] == list[j + 1][0])
            {
                int k = 1;
                while (list[j][k] == list[j + 1][k] && (list[j][k] != '\0' &&
list[j + 1][k] != '\0'))
                {
                    k++;
                }
                if (list[j][k] > list[j + 1][k])
                {
                    swap(&list[j], &list[j + 1]);
                }
            }
        }
    }
}

int main()
{
    char **list = nameList();
    sortList(list);
    displayList(list);
    return 0;
}

```

PTO

OUTPUT

```
Enter the number of students: 4
Enter the number of characters in the name: 4
Enter the name: Andy
Enter the number of characters in the name: 4
Enter the name: Zack
Enter the number of characters in the name: 5
Enter the name: Aandy
Enter the number of characters in the name: 4
Enter the name: Aary
Name list:
Aandy
Aary
Andy
Zack
PS D:\Documents\NIT-K\My_Second_Sem\CS111\Code\M7> |
```