**Delhi Technological University**

Department of Software Engineering



**Data Structures (SE-203)**

**LAB FILE**

**Submitted To :**

Mr. Ankur Narwal

Department of Software Engineering

**Submitted By :**

Aditya Bibhas Sahu

23/SE/009

**Index**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Experiment** | **Date** | **Remarks** |
| 1. | Explore the Linux terminal. Check all basic commands for file  handling, compiling c programs, debugging. Install a Linux OS either  on a virtual machine or dual partition.  Write a small program in C to reverse an array, compile using  c and generate a valid output file. | 21/08/24 |  |
| 2. | Create a menu driver program that will take input from the user to :   1. Enter elements in a one dimensional array. 2. Delete element in a one dimensional array (have all   conditions, beginning, last, middle index).   1. Find the largest element . 2. Find the smallest element . | 04/09/24 |  |
| 3. | Write a menu driven program to  l. Merge two strings  2 reverse strings  Find a substring and replace it with another string.  All inputs to be taken from the user. | 09/10/24 |  |
| 4. |  |  |  |
| 5. |  |  |  |
| 6. |  |  |  |
| 7. |  |  |  |
| 8. |  |  |  |
| 9. |  |  |  |
| 10. |  |  |  |

**Experiment-1**

**Aim:** Explore the Linux terminal. Check all basic commands for file handling, compiling c programs, debugging. Install a Linux OS either on a virtual machine or dual partition.

Write a small program in C to reverse an array, compile using c and generate a valid output file.

**Code:**

#include<stdio.h>

// Q1 : Reverse an array

void main(){

int arr[] = {1,23,34,2,112,10,21};

int size=sizeof(arr)/sizeof(arr[0]);

for(int i=0;i<size/2;i++){

int temp=arr[i];

arr[i]=arr[size-i-1];

arr[size-i-1]=temp;

}

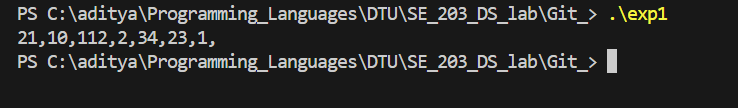
for(int i=0;i<size;i++){

printf("%d,",arr[i]);

}

}

**Output:**



**Experiment-2**

**Aim:** Create a menu driver program that will take input from the user to :

1. Enter elements in a one dimensional array.
2. Delete element in a one dimensional array (have all conditions, beginning,

last, middle index).

1. Find the largest element .
2. Find the smallest element .

**Code:**

#include<stdio.h>

void main(){

printf("Please Enter len of array\n");

int len;

scanf("%d",&len);

int arr[len+1];

int size=0;

int flag=1;

while(flag){

printf("Please Select an input \n");

printf("1. Enter elements in a one dimensional array \n");

printf("2. delete element in a one dimensional array (have all conditions, beginning, last, middle index) \n");

printf("3, Find the largest element \n");

printf("4. Find the smallest element \n");

printf("5. To exit \n");

int a;

scanf("%d",&a);

if(a==1){

printf("Please Enter idx \n");

int idx,num;

scanf("%d",&idx);

printf("Please Enter num \n");

scanf("%d",&num);

for(int i=0;i<size+1;i++){

if(i>=idx){

int temp=arr[i];

arr[i]=num;

num=temp;

}

}

size++;

}

else if(a==2){

printf("Please Enter idx \n");

int idx;

scanf("%d",&idx);

for(int i=0;i<size-1;i++){

if(i>=idx){

arr[i]=arr[i+1];

}

}

size--;

}

else if(a==3){

int max=-1e7;

for(int i=0;i<size;i++){

if(arr[i]>max) max=arr[i];

}

printf("Max = %d\n",max);

}

else if(a==4){

int min=1e7;

for(int i=0;i<size;i++){

if(arr[i]<min) min=arr[i];

}

printf("Min = %d",min);

}

else{

flag=0;

}

for(int i=0;i<size;i++){

printf("%d,",arr[i]);

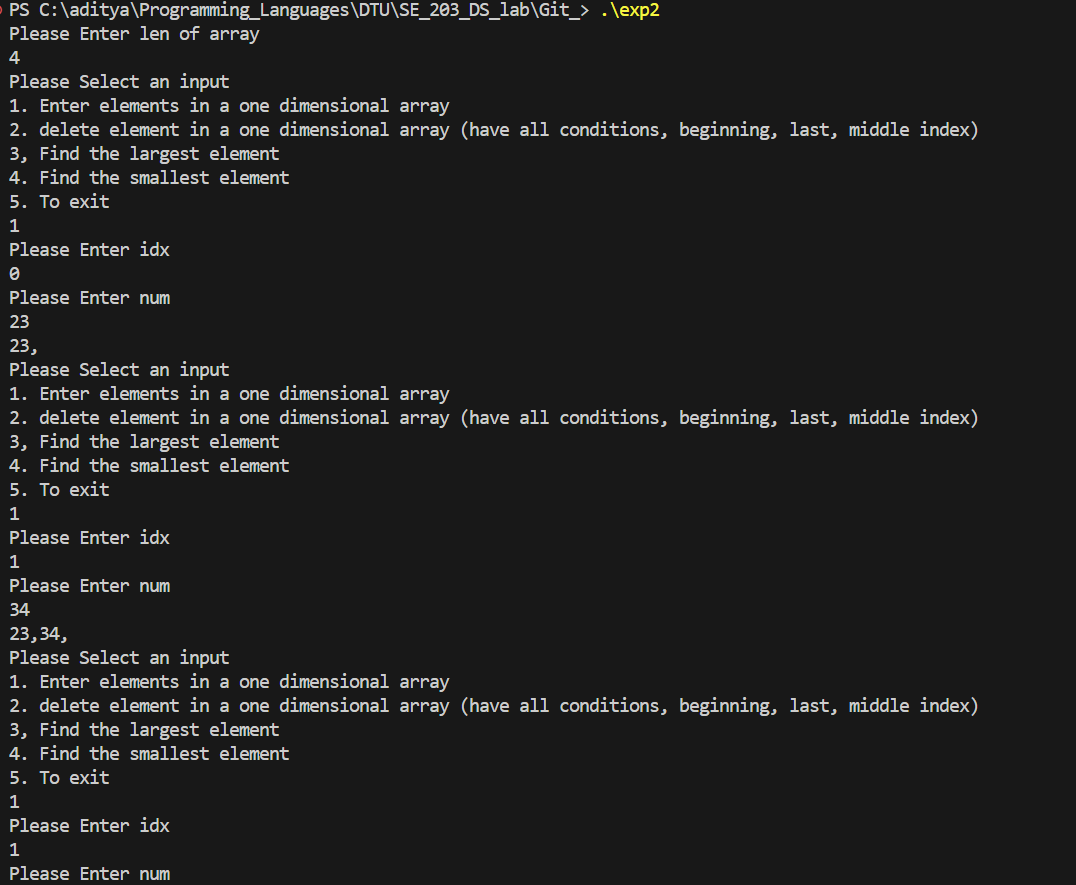
}

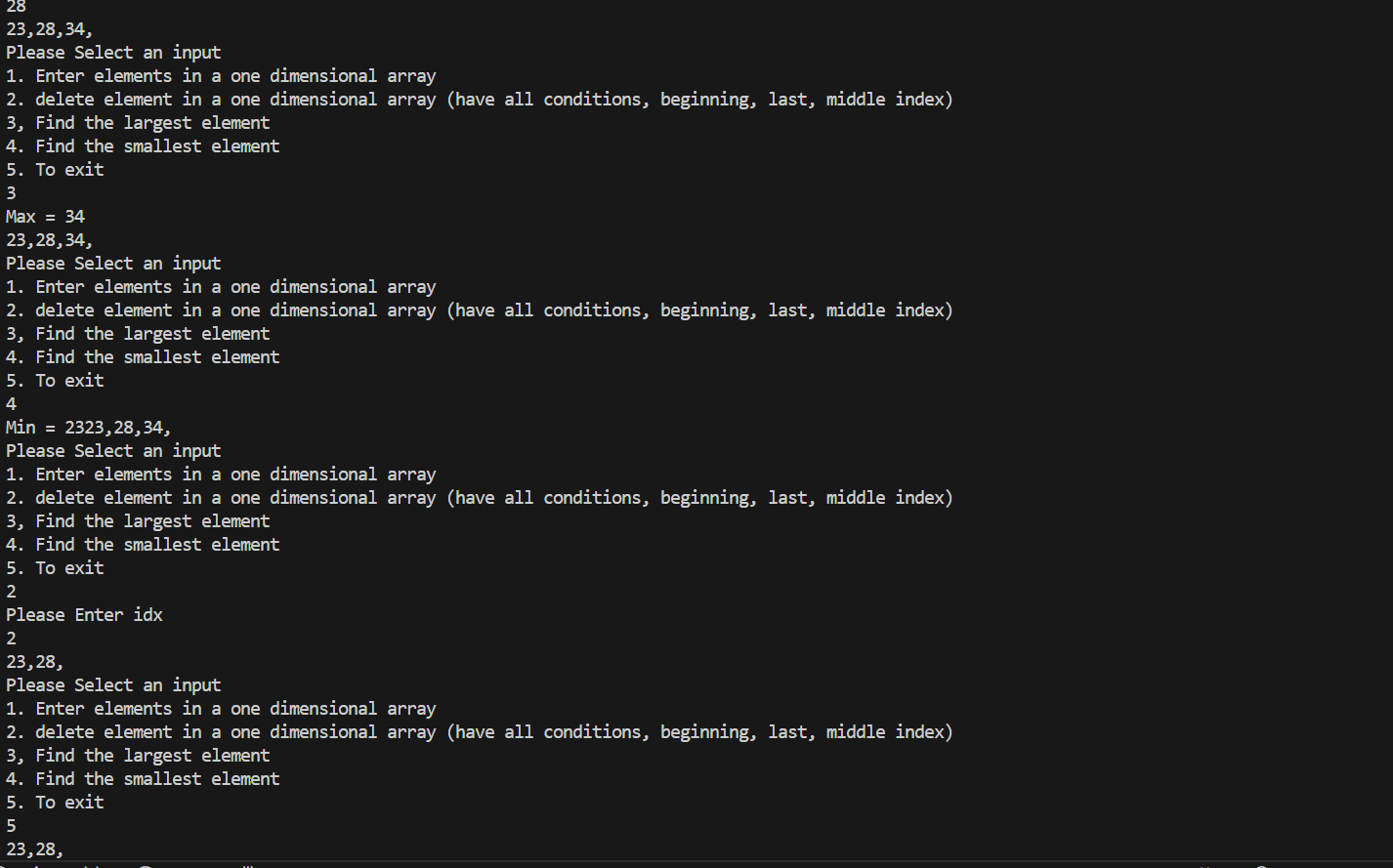
printf("\n");

}

}

**Output:**

****

****

**Experiment-3**

**Aim:** Write a menu driven program to

l. Merge two strings

2 reverse strings

Find a substring and replace it with another string.

All inputs to be taken from the user.

**Code:**

#include <stdlib.h>

#include <stdio.h>

#include <string.h>

char\* merge(char\* str1, char\* str2) {

char\* result = (char\*)malloc(strlen(str1) + strlen(str2) + 1);

strcpy(result, str1);

strcat(result, str2);

return result;

}

char\* reverse(char\* str) {

int len = strlen(str);

for (int i = 0; i < len / 2; i++) {

char temp = str[i];

str[i] = str[len - i - 1];

str[len - i - 1] = temp;

}

return str;

}

char\* substring(char\* str, char\* substr, char\* new\_substr) {

static char buffer[1024];

char\* pos;

if (!(pos = strstr(str, substr))) {

return str;

}

strncpy(buffer, str, pos - str);

buffer[pos - str] = '\0';

strcat(buffer, new\_substr);

strcat(buffer, pos + strlen(substr));

strcpy(str, buffer);

return str;

}

int main() {

int flag = 1;

char str1[100], str2[100], str3[100];

while (flag) {

printf("Please Select an option \n");

printf("1. Merge Two strings \n");

printf("2. Reverse strings\n");

printf("3. Find a substring and replace it with another string\n");

printf("4. Exit \n");

int a;

scanf("%d", &a);

getchar();

printf("\n");

if (a == 1) {

printf("Enter String-1: ");

fgets(str1, sizeof(str1), stdin);

str1[strcspn(str1, "\n")] = 0;

printf("Enter String-2: ");

fgets(str2, sizeof(str2), stdin);

str2[strcspn(str2, "\n")] = 0;

char\* result = merge(str1, str2);

printf("The merged string: %s\n", result);

free(result);

} else if (a == 2) {

printf("Enter string to be reversed: ");

fgets(str3, sizeof(str3), stdin);

str3[strcspn(str3, "\n")] = 0;

printf("Reversed string: %s\n", reverse(str3));

} else if (a == 3) {

printf("Enter the main string: ");

fgets(str1, sizeof(str1), stdin);

str1[strcspn(str1, "\n")] = 0;

printf("Enter the substring to find: ");

fgets(str2, sizeof(str2), stdin);

str2[strcspn(str2, "\n")] = 0;

printf("Enter the replacement string: ");

fgets(str3, sizeof(str3), stdin);

str3[strcspn(str3, "\n")] = 0;

printf("The new string: %s\n", substring(str1, str2, str3));

} else {

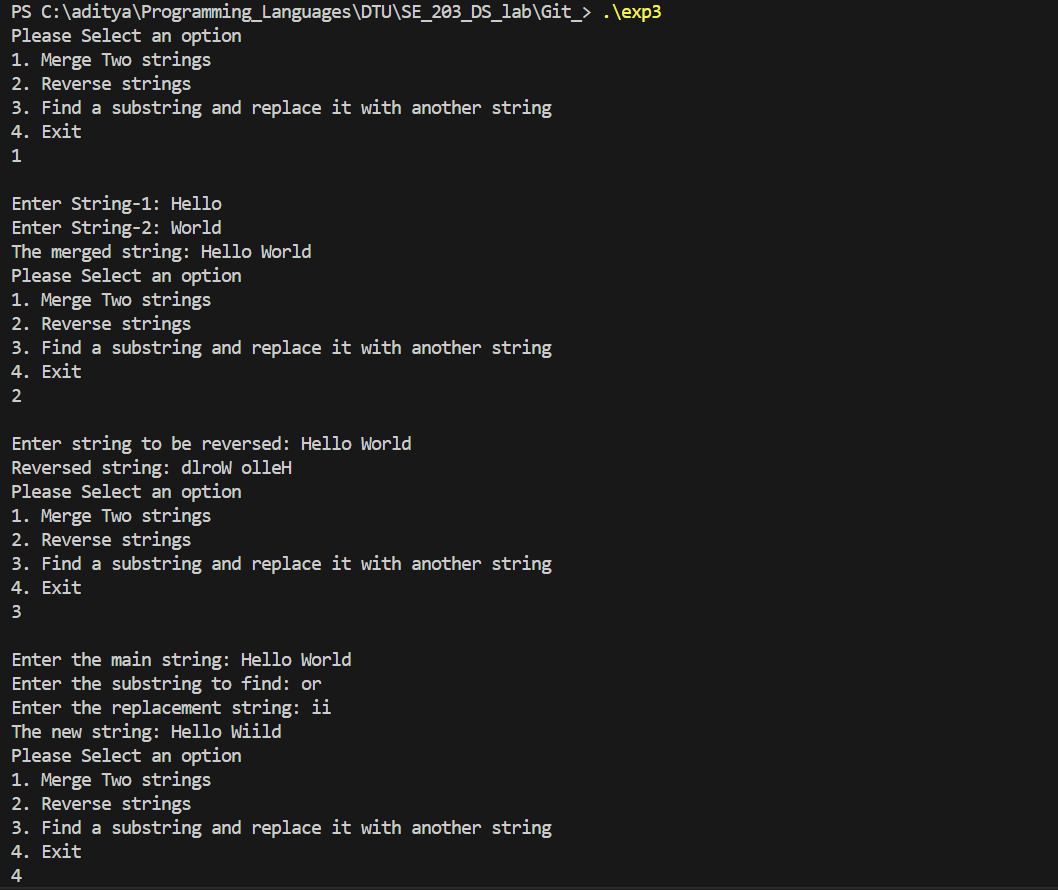
flag = 0;

}

}

}

**Output:**

****