CITY UNIVERCITY

Course Code: **CSE 114**

Course Title: **Structured Programming Language Laboratory**

Submitted to:

**Md. Samrat Ali Abu Kawser**

Lecturer & Coordinator (Evening)

Department of CSE

Submitted by:

**Md. Tahmeedul Islam**

Id: 2125702012

Batch: 57th

Fall 2021

Submission date: 22 February, 2022

LAB REPORT

On Array, Function, String, Pointer and File

**Array**

**1. Use loop to scan & print array elements**

**Answer:** #include<stdio.h>

int main(){

int i;

int iAmAnArray [12];

printf("Enter 12 values: ");

for (i = 0; i < 12; i++){

scanf("%d", &iAmAnArray[i]);

}

for (i = 0; i < 12; i++){

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

}

return 0;

}

**Input:** 1 2 3 4 5 6 7 8 9 0 1 2

**Output:** 1 2 3 4 5 6 7 8 9 0 1 2

**2. Find the sum of two metrics**

**Answer:** #include <stdio.h>

int main(){

int array1[2][2], array2[2][2], resultArray[2][2], i, j, k;

// Int. the value of the first array

printf("Enter value of Matrix 1 (2\*2):\n");

for (i = 0; i < 2; i++){

for (j = 0; j < 2; j++){

scanf("%d", &array1[i][j]);

}

}

// Int. the value of the second array

printf("\nEnter value of Matrix 2 (2\*2):\n");

for (i = 0; i < 2; i++){

for (j = 0; j < 2; j++){

scanf("%d", &array2[i][j]);

}

}

// Calculating the result

printf("\nThe result is:\n");

for (i = 0; i < 2; i++){

for (j = 0; j < 2; j++)

{

resultArray[i][j] = resultArray[i][j] + (array1[i][k] \* array2[k][j]);

}

}

// Printing the result

for (i = 0; i < 2; i++){

for (j = 0; j < 2; j++){

printf("%d \t", resultArray[i][j]);

}

printf("\n");

}

return 0;

}

**Input:** 1111 1111

**Output:** 2222 222

**3. Find the multiplication of two metrics**

**Answer:** #include <stdio.h>

int main(){

int array1[2][2], array2[2][2], resultArray[2][2], i, j, k;

// Int. the value of the first array

printf("Enter value of Matrix 1 (2\*2):\n");

for (i = 0; i < 2; i++){

for (j = 0; j < 2; j++){

scanf("%d", &array1[i][j]);

}

}

// Int. the value of the second array

printf("\nEnter value of Matrix 2 (2\*2):\n");

for (i = 0; i < 2; i++){

for (j = 0; j < 2; j++){

scanf("%d", &array2[i][j]);

}

}

// Calculating the result

printf("\nThe result is:\n");

for (i = 0; i < 2; i++){

for (j = 0; j < 2; j++){

resultArray[i][j] = 0;

for (k = 0; k < 2; k++){

resultArray[i][j] = resultArray[i][j] + (array1[i][k] \* array2[k][j]);

}

}

}

// Printing the result

for (i = 0; i < 2; i++){

for (j = 0; j < 2; j++){

printf("%d \t", resultArray[i][j]);

}

printf("\n");

}

return 0;

}

**Input:** 1234 5678

**Output:** 19 22 43 50

**4. Find the sum of two array elements**

**Answer:** #include<stdio.h>

void main(){

int i;

int iAmAnArray [12], sum = 0;

printf("Enter 12 values: ");

for (i = 0; i < 12; i++){

scanf("%d", &iAmAnArray[i]);

}

for (i = 0; i < 12; i++){

sum = sum + iAmAnArray[i];

}

printf("Sum is: %d, ", sum);

}

**Input:** 111111111111

**Output:** 12

**5.Find the average of two array elements**

**Answer:** #include<stdio.h>

void main(){

int i;

int iAmAnArray [12], sum = 0, avarage;

printf("Enter 12 values: ");

for (i = 0; i < 12; i++)

scanf("%d", &iAmAnArray[i]);

for (i = 0; i < 12; i++)

sum = sum + iAmAnArray[i];

avarage = sum/12;

printf("Avarage is: %d, ", avarage);

}

**Input:** 111111111111

**Output:** 1

**6. Write a C program to read and print elements of array. using recursion**

**Answer:** #include <stdio.h>

void print(int array[], int start, int arraySize)

{

if(start == arraySize){

return;

}else{

printf("%d ", array[start]);

print(array, start+1, arraySize);

}

}

void main(){

int array[100], arraySize, i;

printf("Enter the size of the Array: ");

scanf("%d", &arraySize);

printf("Enter %d numbers: ", arraySize);

for (i = 0; i < arraySize; i++){

scanf("%d", &array[i]);

}

printf("The array is: ");

print(array, 0, arraySize);

}

**Input:** 123456

**Output:** 123456

**7. Write a C program to print all negative elements in an array**

**Answer:** #include <stdio.h>

void main() {

int array[5], i;

printf("Enter 5 numbers: ");

for (i = 0; i < 5; i++){

scanf("%d", &array[i]);

}

printf("Negetive numbers are: ");

for (i = 0; i < 5; i++){

if (array[i] < 0){

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

}

}

}

**Input:** 12 -5 6 -7

**Output:** -5 -7

**8. Write a C program to find sum of all array elements. - using recursion**

**Answer:** #include<stdio.h>

int result(int array[], int start, int arraySize){

if (start == arraySize){

return 0;

}else{

return array[start] + result(array, start+1, arraySize);

}

}

void main(){

int array[5], i;

printf("Enter 5 numbers: ");

for (i = 0; i < 5; i++){

scanf("%d", &array[i]);

}

printf("The result is: %d", result(array, 0, 5));

}

**Input:** 1 2 3 4 5

**Output:** 15

**9. Write a C program to find maximum and minimum element in an array. - using recursion**

**Answer:** #include<stdio.h>

int maximumNumber(int array[], int lastIndexNumber, int max){

if(lastIndexNumber == 0){

return max;

}else{

if(array[lastIndexNumber] > max){

max = array[lastIndexNumber];

}

return maximumNumber(array, lastIndexNumber-1, max);

}

}

int manimumNumber(int array[], int lastIndexNumber, int min){

if(lastIndexNumber == 0){

return min;

}else{

if(array[lastIndexNumber] < min){

min = array[lastIndexNumber];

}

return manimumNumber(array, lastIndexNumber-1, min);

}

}

void main(){

int array[10], arraySize, i, firstElement;

printf("Enter the array size: ");

scanf("%d", &arraySize);

printf("Enter some numbers: ");

for (i = 0; i < arraySize; i++){

scanf("%d", &array[i]);

}

firstElement = array[0];

printf("The maximum number is: %d\n", maximumNumber(array, arraySize-1, firstElement));

printf("The minimum number is: %d\n", manimumNumber(array, arraySize-1, firstElement));

}

**Input:** 1 2 3 4 5 6

**Output:** 1 6

**Function**

**10. Find the sum of two numbers using function**

**Answer:** #include<stdio.h>

int Sum(int a, int b) {

return a + b;

}

void main() {

int number1, number2;

printf("Please enter any two number: ");

scanf("%d %d", &number1, &number2);

printf("Result: %d", Sum(number1, number2));

}

**Input:** 1 2

**Output:** 3

**11. Print Fibonacci series to n**

**Answer:** #include <stdio.h>

int n1 = 0, n2 = 1, n3;

void fibonacci(int n){

if (n > 0){

n3 = n1 + n2;

n1 = n2;

n2 = n3;

printf("%d ", n3);

fibonacci(n - 1);

}

}

void main(){

int userInput;

printf("How many fibonacci number you want to print: ");

scanf("%d", &userInput);

printf("0 ");

fibonacci(userInput - 1);

}

**Input:** 5

**Output:** 0 1 2 3 5

**12. Write a C program to find cube of any number using function**

**Answer:** #include<stdio.h>

#include<math.h>

int cube(int n){

return pow(n, 3);

}

void main(){

int userInput;

printf("Please enter any number to find the cube: ");

scanf("%d", &userInput);

printf("The cube of %d is: %d", userInput, cube(userInput));

}

**Input:** 3

**Output:** 27

**13. Write a C program to find diameter, circumference and area of circle using functions**

**Answer:** #include <stdio.h>

#define PI 3.1416

float diameter(int radius){

return radius + radius;

}

float circumference(int radius){

return 2 \* PI \* radius;

}

float area(int radius){

return PI \* radius \* radius;

}

void main() {

int radius;

printf("Enter the radius of a triangle: ");

scanf("%d", &radius);

printf("\nThe diameter is : %.2f", diameter(radius));

printf("\nThe circumference is : %.2f", circumference(radius));

printf("\nThe area is : %.2f", area(radius));

}

**14. Write a C program to find maximum and minimum between two numbers using functions**

**Answer:** #include <stdio.h>

int max(int a, int b){

if (a > b) return a;

else return b;

}

int min(int a, int b){

if (a > b) return b;

else return a;

}

void main(){

int number1, number2;

printf("Please enter two numbers: ");

scanf("%d%d", &number1, &number2);

printf("The maximum is: %d\n", max(number1, number2));

printf("The manimum is: %d", min(number1, number2));

}

**Input:** 12 65

**Output:** 65

**15. Write a C program to check whether a number is even or odd using functions**

**Answer:** #include <stdio.h>

void checkEvenOrOdd(int n){

if (n % 2 == 0)

printf("%d is an Even number.", n);

else

printf("%d is a Odd number.", n);

}

void main(){

int userInput;

printf("Please enter any number: ");

scanf("%d", &userInput);

checkEvenOrOdd(userInput);

}

**Input:** 12

**Output:** 12 is an odd number

**String**

**16. Use of puts and gets**

**Answer:** #include<stdio.h>

void main(){

char name[30];

printf("Enter your name: ");

gets(name);

printf("Your name is: ");

puts(name);

}

**17. Use of fgets() and strcpy()**

**Answer:** #include<stdio.h>

#include<string.h>

void main(){

char string1[30], string2[30];

printf("Enter string1: ");

fgets(string1, sizeof(string1), stdin);

printf("Enter string2: ");

fgets(string2, sizeof(string2), stdin);

strcpy(string1, string2);

puts(string1);

}

**18. Use of strcmp()**

**Answer:** #include<stdio.h>

#include<string.h>

void main(){

char string1[30], string2[30];

printf("Enter string1: ");

fgets(string1, sizeof(string1), stdin);

printf("Enter string2: ");

fgets(string2, sizeof(string2), stdin);

printf("The result is: %d", strcmp(string1, string2));

}

**19. Use of strcat()**

**Answer:** #include<stdio.h>

#include<string.h>

void main(){

char firstName[30], lastName[30];

printf("Enter firstName: ");

fgets(firstName, sizeof(firstName), stdin);

printf("Enter lastName: ");

fgets(lastName, sizeof(lastName), stdin);

printf("The result is: %s", strcat(firstName, lastName));

}

**19. Write a C program to find length of a string (Without function)**

**Answer:** #include<stdio.h>

void main(){

int i, count = 0;

char name[30];

printf("Enter your name: ");

scanf("%s", name);

for (i = 0; name[i] != '\0' ; i++){

count++;

}

printf("Total character is: %d", count);

}

**20. Copy one string to another (Without function)**

**Answer:** #include <stdio.h>

void main() {

char string1[30];

char string2[30];

int i;

printf("Enter the first string: ");

gets(string1);

for(i=0; string1[i]!='\0'; i++){

string2[i] = string1[i];

}

string2[i] = '\0';

printf("First string is %s\n", string1);

printf("Second string is %s", string2);

}

**Pointer**

**21. Pointer syntax**

**Answer:** #include<stdio.h>

void main(void){

int v=100, \*p;

p = &v;

printf("%d\n", &v); // Location of v

printf("%d\n", v); // Value of v

printf("%d\n", p); // Location of v which is stored in p

printf("%d\n", \*p); // Location value of v which is stored in p

printf("%d\n", &p); // Location of p

}

**22. Use of pointer**

**Answer:** #include<stdio.h>

void main(){

int v=3, \*p;

p=&v;

printf("Address of V is %u", &v);

printf("\nAddress of P is %u", &p);

printf("\nValue of P is %u", p);

printf("\nValue of V is %u", v);

printf("\nValue of V is %u", \*(&v));

printf("\nValue of V is %u", \*p);

}

**23. Find the out of**

#include<stdio.h>

void main(void){

int x=2, y=3, \*p, \*q;

p = &x;

q = &y;

p=q;

printf("%d %d %d %d", x, y, \*p, \*q);

\*p=3;

\*q=4;

x=y;

printf("\n%d %d %d %d", x, y, \*p, \*q);

}

**Answer:** 2 3 3 3

4 4 4 4

**24.** **Function with call by value**

**Answer:** #include<stdio.h>

void swap(int a, int b);

void main(void){

int x=2, y=3;

swap(x, y);

}

void swap(int a, int b){

int temp=a;

a = b;

b = temp;

printf("%d %d", a, b);

}

**25. Function with call by reference**

**Answer:** #include<stdio.h>

void swap(int \*a, int \*b);

void main(void){

int x=2, y=3;

swap(&x, &y);

}

void swap(int \*a, int \*b){

int temp=\*a;

\*a = \*b;

\*b = temp;

printf("%d %d", \*a, \*b);

}

**26. Find are and radius (Using call by reference)**

**Answer:** #include<stdio.h>

void calculation(int r, float \*a, float \*p);

void main(void){

int radius;

float area, perimeter;

printf('Please enter the radius: ');

scanf("%d", &radius);

calculation(radius, &area, &perimeter);

printf("Area is %.2f", area);

printf("Perimeter is %.2f", perimeter);

}

void calculation(int r, float \*a, float \*p){

\*a=3.1416\*r\*r;

\*p=2\*3.1416\*r;

}

**File**

**26. Create and close a File**

**Answer:** #include<stdio.h>

void main(){

FILE \*mFile;

mFile = fopen("MyFile.txt","w");

if(mFile == NULL){

printf("File doesn't exist.");

}else{

printf("File is oppened");

fclose(mFile);

}

}

**27. Use of fputc()**

**Answer:** #include<stdio.h>

#include<string.h>

void main(){

FILE \*mFile;

char name[30] = "Tahmeedul Islam";

int nameLegnth = strlen(name);

int i;

mFile = fopen("MyFile.txt", "w");

if (mFile != NULL){

for (i = 0; i < nameLegnth; i++){

fputc(name[i], mFile);

// What to write, Where to write

}

fclose(mFile);

}else{

printf("File not found");

}

}

**28. Use of fputs()**

**Answer:** #include<stdio.h>

void main(){

char userName[30];

FILE \*mFile = fopen("MyFile.txt", "a");

if (mFile != NULL){

printf("Please enter your name: ");

gets(userName);

fputs(userName, mFile);

fputs("\n", mFile);

printf("Name saved");

fclose(mFile);

}else{

printf("Unable to open file");

}

}

**29. Use of fprintf()**

**Answer:** #include <stdio.h>

void main(){

char name[20], id[10], batch[5];

FILE \*mFile = fopen("MyFile.txt", "a");

if (mFile != NULL){

printf("Enter your name: ");

gets(name);

printf("Enter your id: ");

gets(id);

printf("Enter your batch: ");

gets(batch);

fprintf(mFile, "\n\nName: %s\nId: %s\nBatch: %sth", name, id, batch);

printf("Data saved");

fclose(mFile);

}else{

printf("File not found");

}

}

**30. Use of fgetc()**

**Answer:** #include <stdio.h>

void main(){

char ch;

FILE \*mFile = fopen("MyFile.txt", "r");

if (mFile != NULL){

printf("File oppend: ");

while (!feof(mFile)){

ch = fgetc(mFile);

printf("%c", ch);

}

fclose(mFile);

}else{

printf("File not found");

}

}

**31. Use of fgets()**

**Answer:** #include <stdio.h>

void main(){

char ch[20];

FILE \*mFile = fopen("MyFile.txt", "r");

if (mFile != NULL){

printf("File oppend: \n");

while (!feof(mFile)){

fgets(ch, 10, mFile);

printf("%s", ch);

}

fclose(mFile);

}else{

printf("File not found");

}

}

**32. Use of fscanf()**

**Answer:** #include<stdio.h>

void main() {

char ch[10], ch2[10];

FILE \*mFile = fopen("MyFile.txt", "r");

if (mFile != NULL){

fscanf(mFile, "%s %s", ch, ch2);

printf("%s %s", ch, ch2);

fclose(mFile);

}else{

printf("File not found");

}

}