**Passing array to function - 1**

**C program to pass a single element of an array to function**

#include <stdio.h>

void display(int age)

{

printf("%d", age);

}

int main()

{

int ageArray[] = { 2, 3, 4 };

display(ageArray[2]); //Passing array element ageArray[2] only.

return 0;

}

**Output**

4

**Passing array to function - 2**

As we already know in this type of function call, the actual parameter is copied to the formal parameters.

#include <stdio.h>

void disp( char ch)

{

printf("%c ", ch);

}

int main()

{

char arr[] = {'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j'};

for (int x=0; x<10; x++)

{

/\* I’m passing each element one by one using subscript\*/

disp (arr[x]);

}

return 0;

}

**Output:**

a b c d e f g h i j

**Passing array to function – 3**

**Passing an entire one-dimensional array to a function**

While passing arrays as arguments to the function, only the name of the array is passed (,i.e, starting address of memory area is passed as argument).

**C program to pass an array containing age of person to a function. This function should find average age and display the average age in main function.**

#include <stdio.h>

float average(float age[]);

int main()​

{

float avg, age[] = { 23.4, 55, 22.6, 3, 40.5, 18 };

avg = average(age); /\* Only name of array is passed as argument. \*/

printf("Average age=%.2f", avg);

return 0;

}

float average(float age[])

{

int i;

float avg, sum = 0.0;

for (i = 0; i < 6; ++i) {

sum += age[i];

}

avg = (sum / 6);

return avg;

}

**Passing array to function - 4**

#include<stdio.h>

int minarray(int arr[],int size){

int min=arr[0];

int i=0;

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

if(min>arr[i]){

min=arr[i];

}

}//end of for

return min;

}//end of function

int main(){

int i=0,min=0;

int numbers[]={4,5,7,3,8,9};//declaration of array

min=minarray(numbers,6);//passing array with size

printf("minimum number is %d \n",min);

return 0;

}

Output

minimum number is 3

**2D Array – 1**

#include<stdio.h>

int main(){

/\* 2D array declaration\*/

int disp[2][3];

/\*Counter variables for the loop\*/

int i, j;

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

for(j=0;j<3;j++) {

printf("Enter value for disp[%d][%d]:", i, j);

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

}

}

//Displaying array elements

printf("Two Dimensional array elements:\n");

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

for(j=0;j<3;j++) {

printf("%d ", disp[i][j]);

if(j==2){

printf("\n");

}

}

}

return 0;

}

Output:

Enter value for disp[0][0]:1

Enter value for disp[0][1]:2

Enter value for disp[0][2]:3

Enter value for disp[1][0]:4

Enter value for disp[1][1]:5

Enter value for disp[1][2]:6

Two Dimensional array elements:

1 2 3

4 5 6

**2D Array – 2**

// C program to store temperature of two cities for a week and display it.

#include <stdio.h>

const int CITY = 2;

const int WEEK = 7;

int main()

{

int temperature[CITY][WEEK];

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

for(int j = 0; j < WEEK; ++j) {

printf("City %d, Day %d: ", i+1, j+1);

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

}

}

printf("\nDisplaying values: \n\n");

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

for(int j = 0; j < WEEK; ++j)

{

printf("City %d, Day %d = %d\n", i+1, j+1, temperature[i][j]);

}

}

return 0;

}

**Output**

City 1, Day 1: 33

City 1, Day 2: 34

City 1, Day 3: 35

City 1, Day 4: 33

City 1, Day 5: 32

City 1, Day 6: 31

City 1, Day 7: 30

City 2, Day 1: 23

City 2, Day 2: 22

City 2, Day 3: 21

City 2, Day 4: 24

City 2, Day 5: 22

City 2, Day 6: 25

City 2, Day 7: 26

Displaying values:

City 1, Day 1 = 33

City 1, Day 2 = 34

City 1, Day 3 = 35

City 1, Day 4 = 33

City 1, Day 5 = 32

City 1, Day 6 = 31

City 1, Day 7 = 30

City 2, Day 1 = 23

City 2, Day 2 = 22

City 2, Day 3 = 21

City 2, Day 4 = 24

City 2, Day 5 = 22

City 2, Day 6 = 25

City 2, Day 7 = 26

**2D Array – 3**

### Sum of two matrices using Two dimensional arrays

**C program to find the sum of two matrices of order 2\*2 using multidimensional arrays.**

#include <stdio.h>

int main()

{

float a[2][2], b[2][2], c[2][2];

int i, j;

// Taking input using nested for loop

printf("Enter elements of 1st matrix\n");

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

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

{

printf("Enter a%d%d: ", i+1, j+1);

scanf("%f", &a[i][j]);

}

// Taking input using nested for loop

printf("Enter elements of 2nd matrix\n");

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

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

{

printf("Enter b%d%d: ", i+1, j+1);

scanf("%f", &b[i][j]);

}

// adding corresponding elements of two arrays

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

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

{

c[i][j] = a[i][j] + b[i][j];

}

// Displaying the sum

printf("\nSum Of Matrix:");

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

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

{

printf("%.1f\t", c[i][j]);

if(j==1)

printf("\n");

}

return 0;

}

**Ouput**

Enter elements of 1st matrix

Enter a11: 2;

Enter a12: 0.5;

Enter a21: -1.1;

Enter a22: 2;

Enter elements of 2nd matrix

Enter b11: 0.2;

Enter b12: 0;

Enter b21: 0.23;

Enter b22: 23;

Sum Of Matrix:

2.2 0.5

-0.9 25.0

**3D Array – 1**

### Example 3: Three Dimensional Array

**C Program to store values entered by the user in a three-dimensional array and display it.**

#include <stdio.h>

int main()

{

// this array can store 12 elements

int i, j, k, test[2][3][2];

printf("Enter 12 values: \n");

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

for (j = 0; j < 3; ++j) {

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

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

}

}

}

// Displaying values with proper index.

printf("\nDisplaying values:\n");

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

for (j = 0; j < 3; ++j) {

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

printf("test[%d][%d][%d] = %d\n", i, j, k, test[i][j][k]);

}

}

}

return 0;

}

**Output**

Enter 12 values:

1

2

3

4

5

6

7

8

9

10

11

12

Displaying Values:

test[0][0][0] = 1

test[0][0][1] = 2

test[0][1][0] = 3

test[0][1][1] = 4

test[0][2][0] = 5

test[0][2][1] = 6

test[1][0][0] = 7

test[1][0][1] = 8

test[1][1][0] = 9

test[1][1][1] = 10

test[1][2][0] = 11

test[1][2][1] = 12

**Passing array to function - 5**

### #Example: Pass two-dimensional arrays to a function

#include <stdio.h>

void displayNumbers(int num[2][2]);

int main()

{

int num[2][2], i, j;

printf("Enter 4 numbers:\n");

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

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

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

// passing multi-dimensional array to displayNumbers function

displayNumbers(num);

return 0;

}

void displayNumbers(int num[2][2])

{

// Instead of the above line,

// void displayNumbers(int num[][2]) is also valid

int i, j;

printf("Displaying:\n");

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

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

printf("%d\n", num[i][j]);

}

**Output**

Enter 4 numbers:

2

3

4

5

Displaying:

2

3

4

5