1. **Dynamic One Dimensional Array.**

#include <stdio.h>

typedef int\* int\_ptr;

int\_ptr init(int size){

int\_ptr a;

a = (int\_ptr) malloc(sizeof(int)\*size);

return a;

}

void display(int \*a, int size){

int i;

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

printf("%d ",\*(a + i));

}

}

void enter\_value(int \*a, int size){

int i = 0;

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

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

scanf("%d", (a + i));

}

}

void main(){

int\_ptr b;

int size;

printf("Enter total number of element you want in your array: ");

scanf("%d",&size);

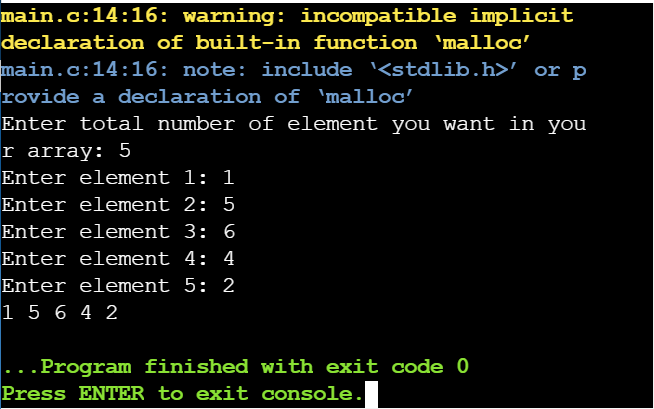
b = init(size);

enter\_value(b,size);

display(b, size);

}

**Output:**



1. **Program to add Dynamic Two Dimensional array in C.**

#include <stdio.h>

#include <stdlib.h>

void display(int\*\*,int, int);

void read(int\*\*,int,int);

void sum(int\*\*,int\*\*,int, int);

int main(){

int row,column, i,j;

int \*\*a,\*\*b,\*\*c;

printf("Enter total number of row and columns: ");

scanf("%d%d",&row,&column);

a = (int\*\*)malloc(sizeof(int\*) \* row);

b = (int\*\*)malloc(sizeof(int\*) \* row);

c = (int\*\*)malloc(sizeof(int\*) \* row);

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

\*(a + i) = (int\*) malloc(sizeof(int)\*column);

\*(b + i) = (int\*) malloc(sizeof(int)\*column);

\*(c + i) = (int\*) malloc(sizeof(int)\*column);

}

read(a,row,column);

display(a,row,column);

read(b,row,column);

display(b,row,column);

sum(a,b,row,column);

return 0;

}

void sum(int \*\*a,int \*\*b,int r, int c){

int i,j;

printf("\nSum is: \n");

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

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

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

}

printf("\n");

}}

void display(int \*\*a, int r, int c){

int i,j;

printf("\nMatrix is: \n");

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

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

printf("%d ",\*(\*(a + i) + j));

}

printf("\n");

}}

void read(int\*\* a, int r, int c){

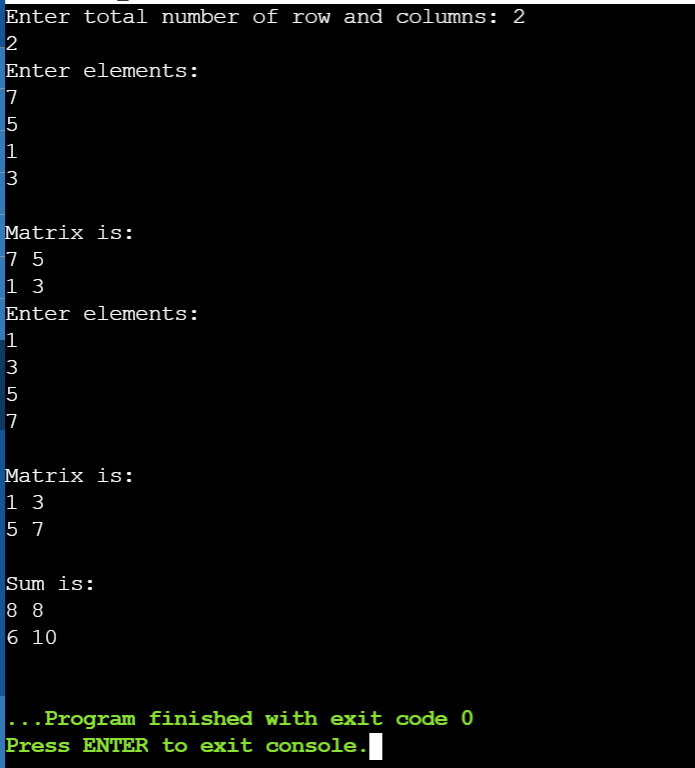
int i,j;

printf("Enter elements:\n");

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

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

scanf("%d",\*(a + i) + j);

}}} **Output:**

1. **Structure within Structure.**

#include <stdio.h>

#include <string.h>

struct college{

char clz\_name[50];

struct Student{

char name[10];

int Class;

} m;

};

int main(){

struct college std;

strcpy(std.clz\_name,"Parlance International Academy.");

strcpy(std.m.name,"Abi");

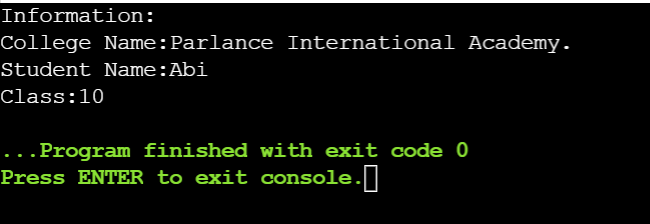
std.m.Class = 10;

printf("Information:\n");

printf("College Name:%s\nStudent Name:%s\nClass:%d",std.clz\_name,std.m.name,std.m.Class);

return 0;

}

**Output:**

1. **Program to Display Names Stored In 2D Array.**

#include <stdio.h>

#include <string.h>

#define ROW 5

#define COLUMN 10

int main(){

char name[ROW][COLUMN]={"TENNIS","FOOTBALL","POLO","VOLLYBALL","CHESS"};

int i,j;

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

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

printf("%c",name[i][j]);

}

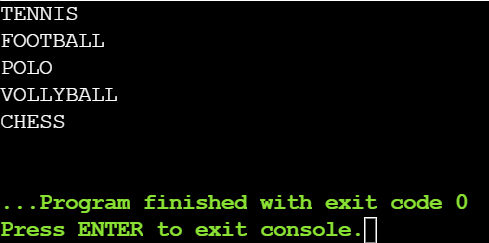
printf("\n");

}

return 0;

}

**Output:**



1. **Passing Structure to the function.**

#include <stdio.h>

#include <string.h>

struct Student{

char name[10];

char grade[3];

int roll;

};

typedef struct Student s;

void display(s m){

printf("\nName:%s\nGrade:%s\nRoll:%d",m.name,m.grade,m.roll);

}

int main(){

s a;

strcpy(a.name,"Abi");

strcpy(a.grade, "A+");

a.roll = 2;

printf("Infomation:");

display(a);

return 0;

}

**output:** 