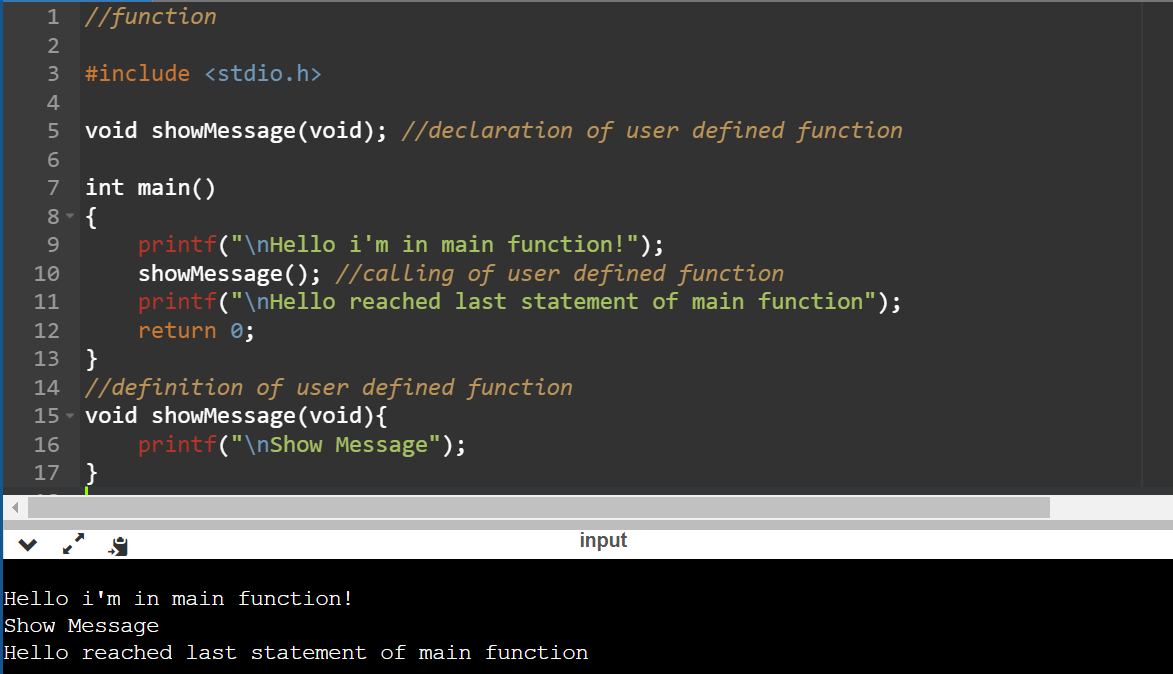
Function-

1)Built in function(printf,etc),

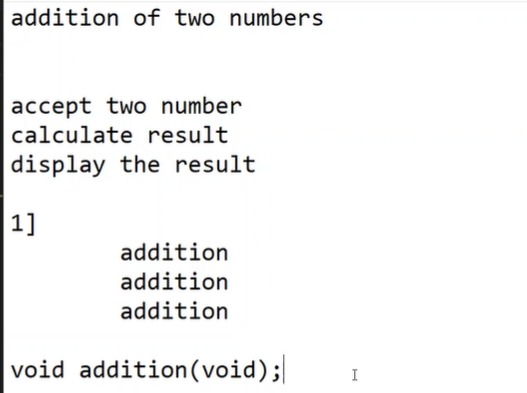
2)User defined function

For function we required 3 things : Declaration,Definition,Calling

**DEMONSTRATION OF FUNCTION:**

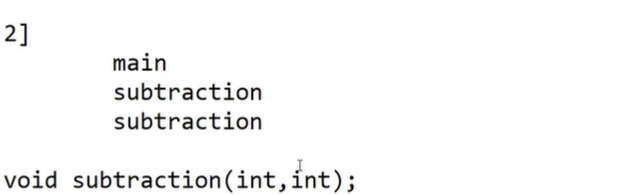


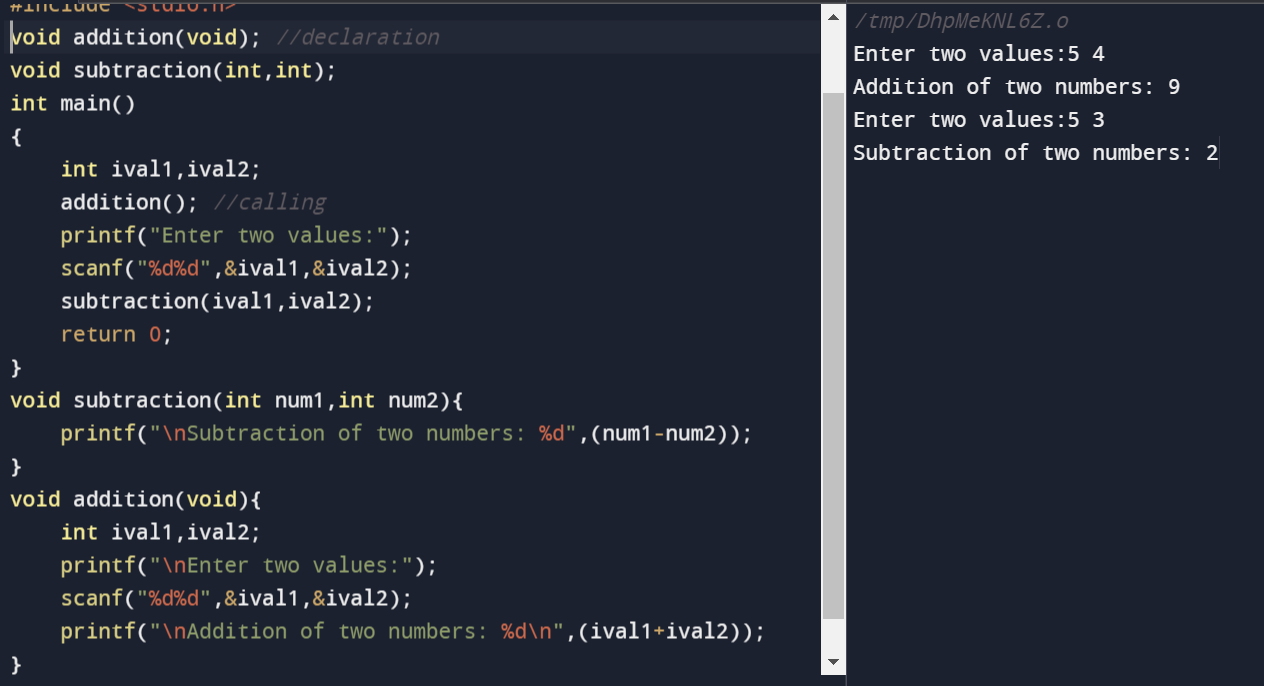
2)

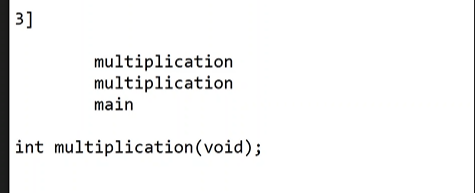


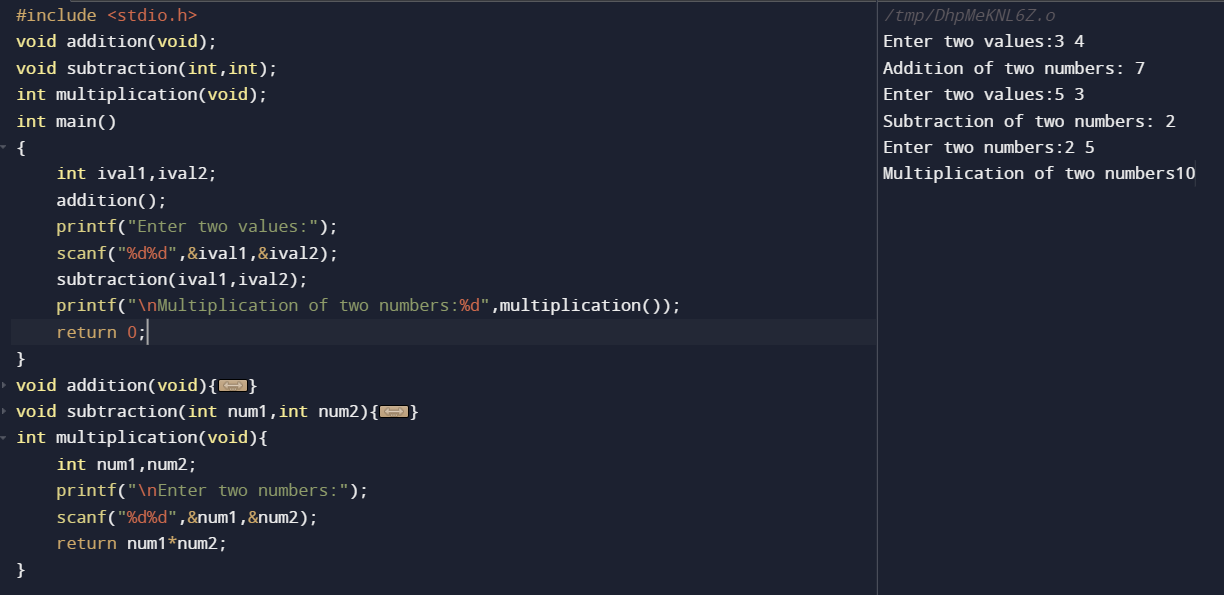


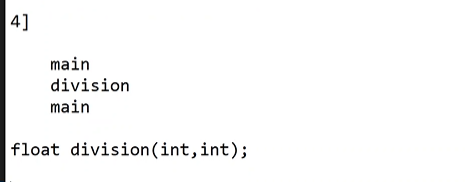
3)

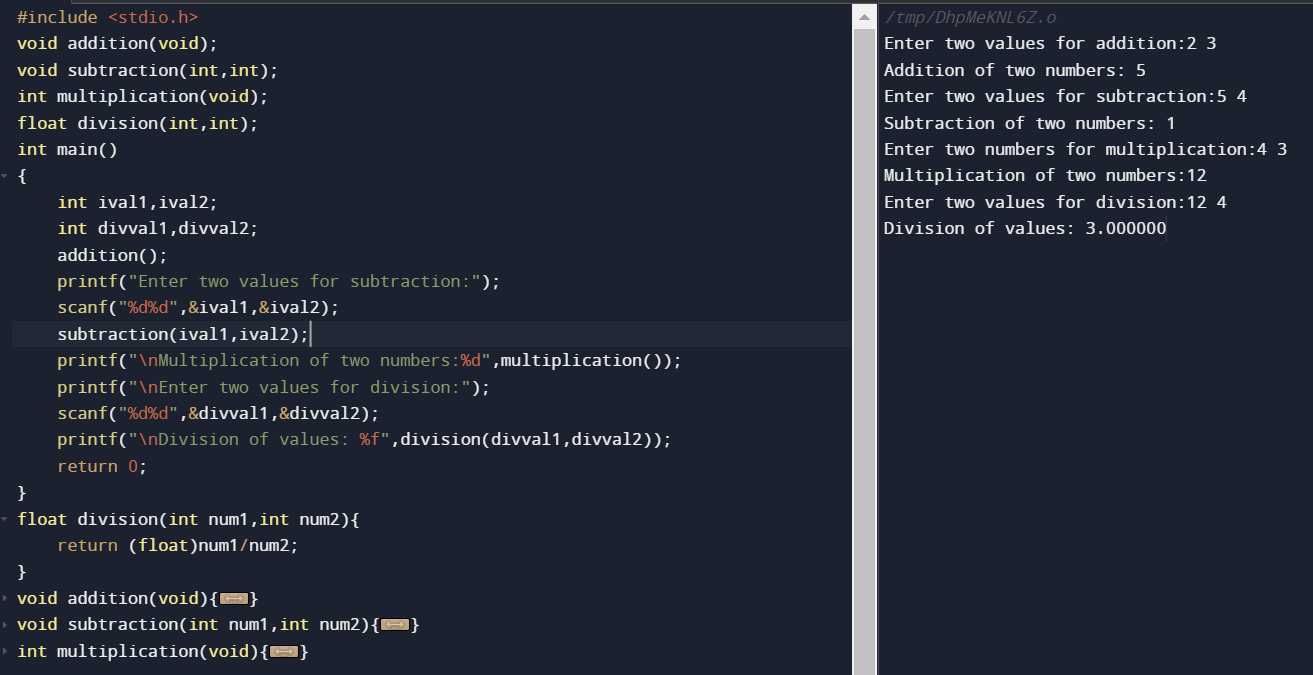












**COMPLETE CODE:**

#include <stdio.h>

void addition(void);

void subtraction(int,int);

int multiplication(void);

float division(int,int);

int main()

{

int ival1,ival2;

int divval1,divval2;

addition();

printf("Enter two values for subtraction:");

scanf("%d%d",&ival1,&ival2);

subtraction(ival1,ival2);

printf("\nMultiplication of two numbers:%d",multiplication());

printf("\nEnter two values for division:");

scanf("%d%d",&divval1,&divval2);

printf("\nDivision of values: %f",division(divval1,divval2));

return 0;

}

float division(int num1,int num2){

return (float)num1/num2;

}

void addition(void){

int ival1,ival2;

printf("\nEnter two values for addition:");

scanf("%d%d",&ival1,&ival2);

printf("\nAddition of two numbers: %d\n",(ival1+ival2));

}

void subtraction(int num1,int num2){

printf("\nSubtraction of two numbers: %d",(num1-num2));

}

int multiplication(void){

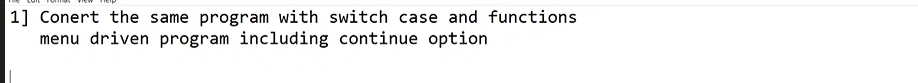
int num1,num2;

printf("\nEnter two numbers for multiplication:");

scanf("%d%d",&num1,&num2);

return num1\*num2;

}



#include <stdio.h>

void addition(void);

void subtraction(void);

void multiplication(void);

void division(void);

int main()

{

int choice;

char ch;

do{

printf("\n1]Addition\n2]Subtraction\n3]Multiplication\n4]Division\n5]Exit\n");

printf("Enter the choice:\n");

scanf("%d",&choice);

switch(choice){

case 1:

addition();

break;

case 2:

subtraction();

break;

case 3:

multiplication();

break;

case 4:

division();

break;

case 5:

printf("Exitting from the code..\n");

break;

default:

printf("Invalid choice\n");

}

}while(choice !=5 );

return 0;

}

void addition(void){

int ival1,ival2;

printf("\nEnter two values for addition:");

scanf("%d%d",&ival1,&ival2);

printf("\nAddition of two numbers: %d\n",(ival1+ival2));

}

void subtraction(void){

int ival1,ival2;

printf("\nEnter two values for subtraction:");

scanf("%d%d",&ival1,&ival2);

printf("\nAddition of two numbers: %d\n",(ival1-ival2));

}void multiplication(void){

int ival1,ival2;

printf("\nEnter two values for multiplication:");

scanf("%d%d",&ival1,&ival2);

printf("\nAddition of two numbers: %d\n",(ival1\*ival2));

}void division(void){

int ival1,ival2;

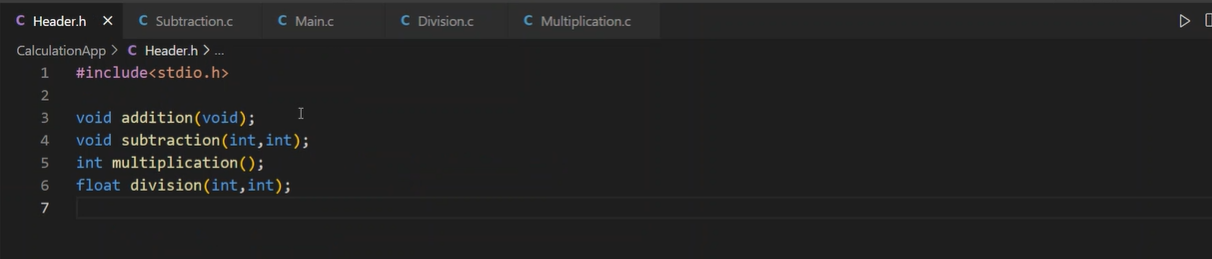
printf("\nEnter two values for division:");

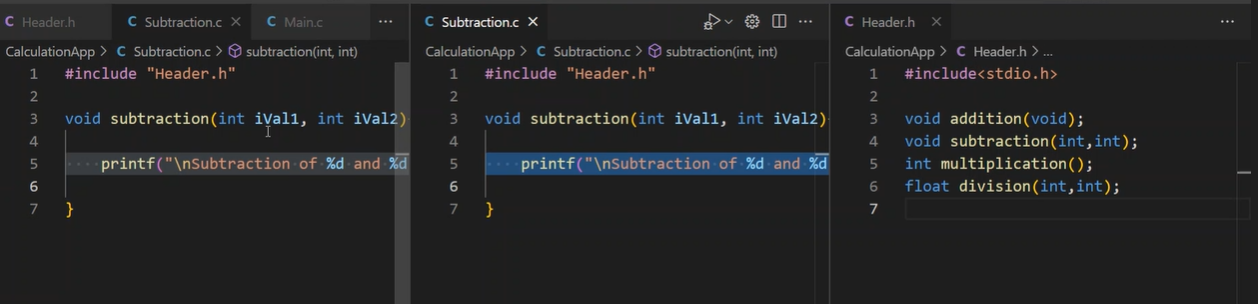
scanf("%d%d",&ival1,&ival2);

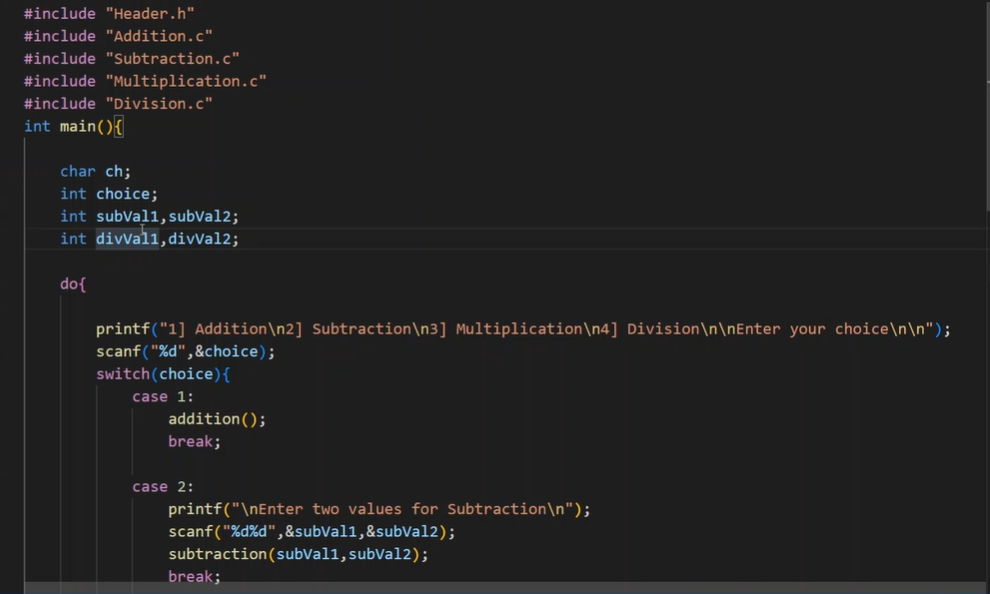
printf("\nAddition of two numbers: %d\n",(ival1/ival2));

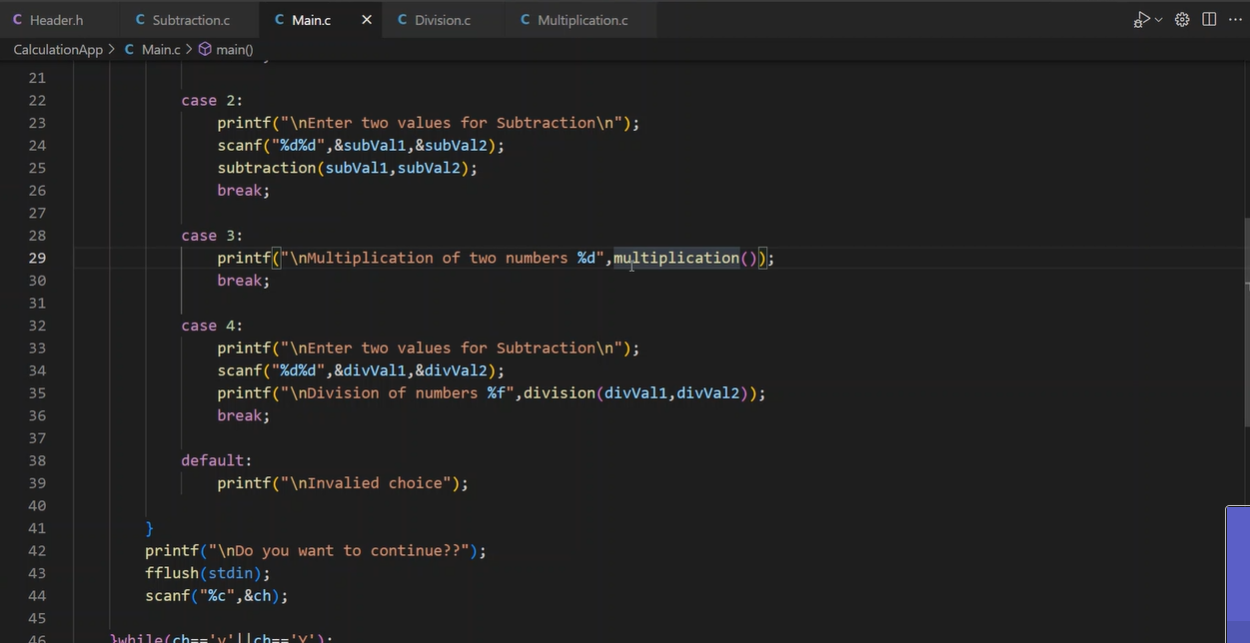
}

**SIR**

****

****

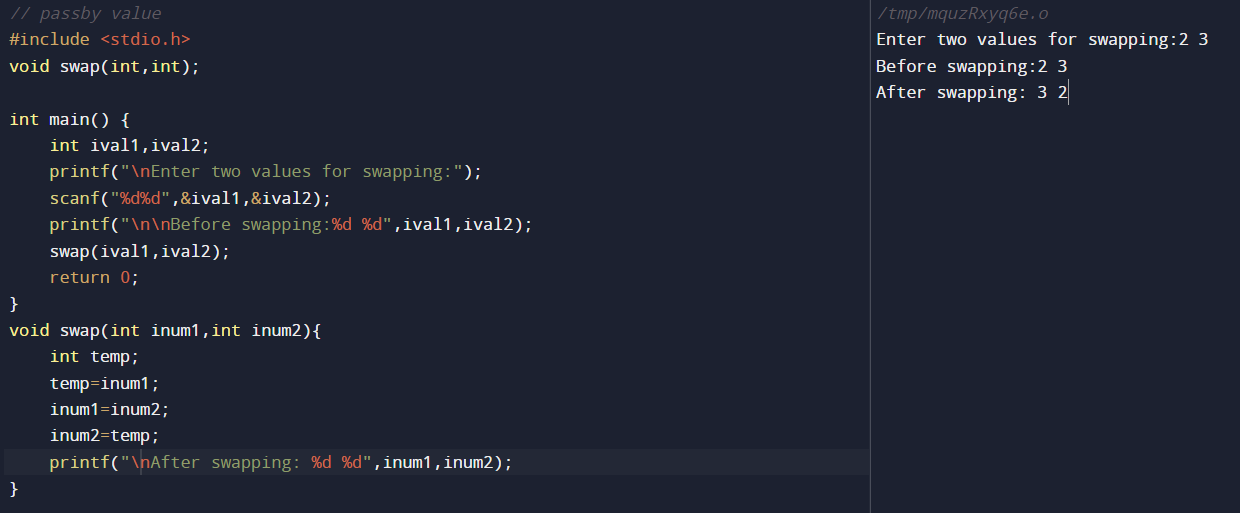
****



**C-**

Pass by value , Pass by address

**PASS BY VALUE:**

****

// passby value

#include <stdio.h>

void swap(int,int);

int main() {

int ival1,ival2;

printf("\nEnter two values for swapping:");

scanf("%d%d",&ival1,&ival2);

printf("\n\nBefore swapping:%d %d",ival1,ival2);

swap(ival1,ival2);

return 0;

}

void swap(int inum1,int inum2){

int temp;

temp=inum1;

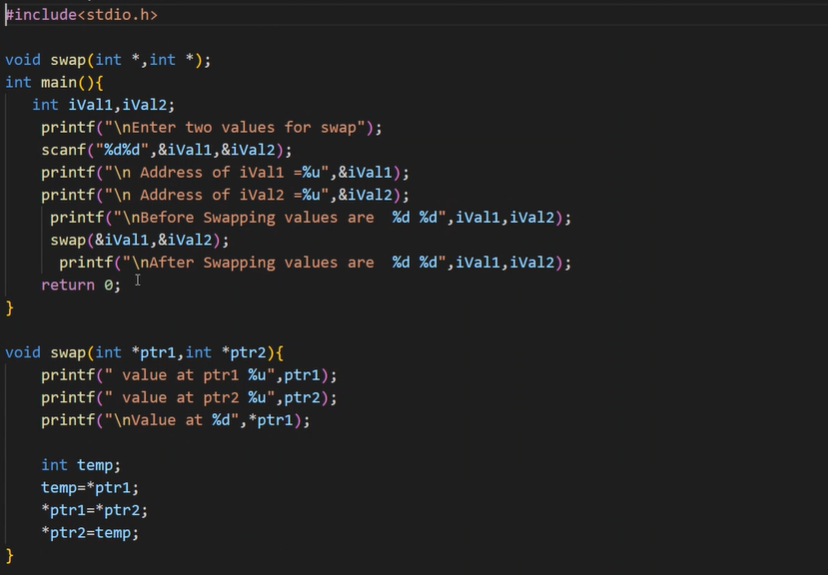
inum1=inum2;

inum2=temp;

printf("\nAfter swapping: %d %d",inum1,inum2);

}

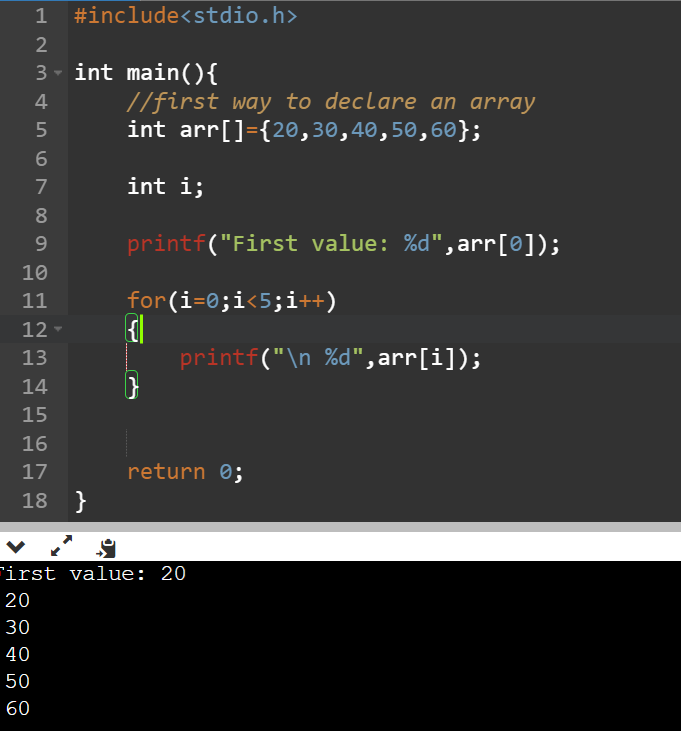
**Pass By address:**



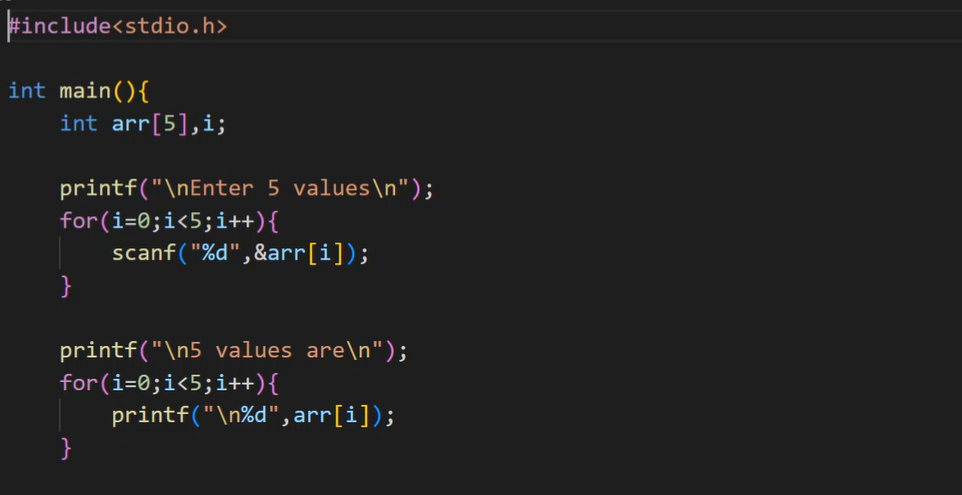
Comment statement will work the same.

ARRAY:

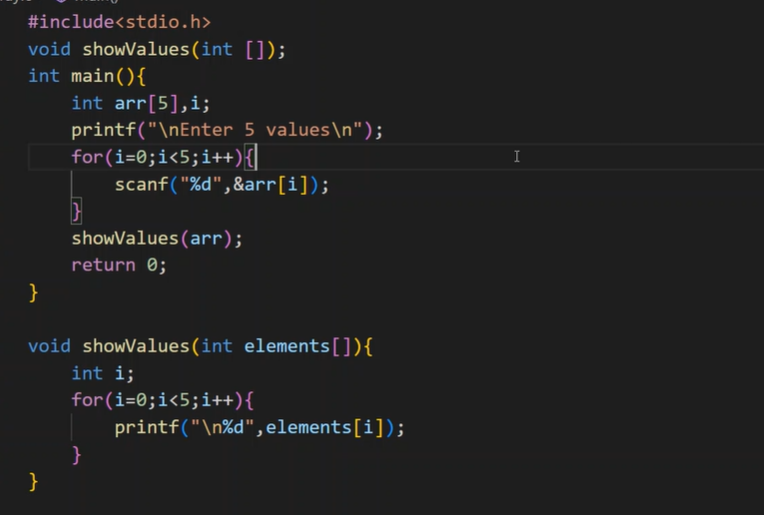
Normal printing:



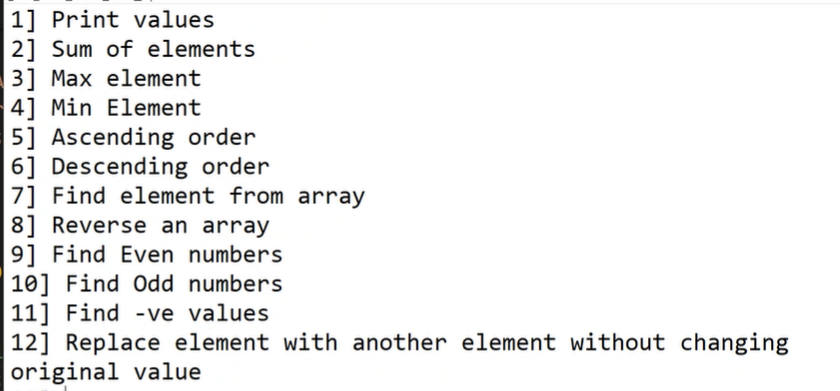
Array elements accepting from user:



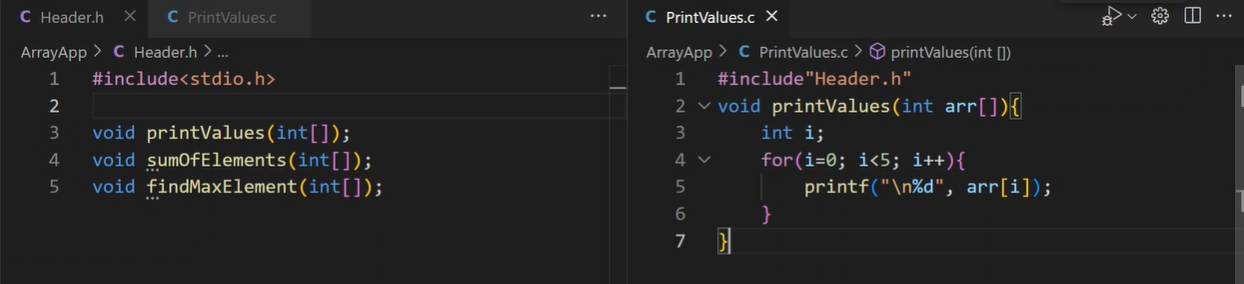
ARRAY WITH FUNCTION:

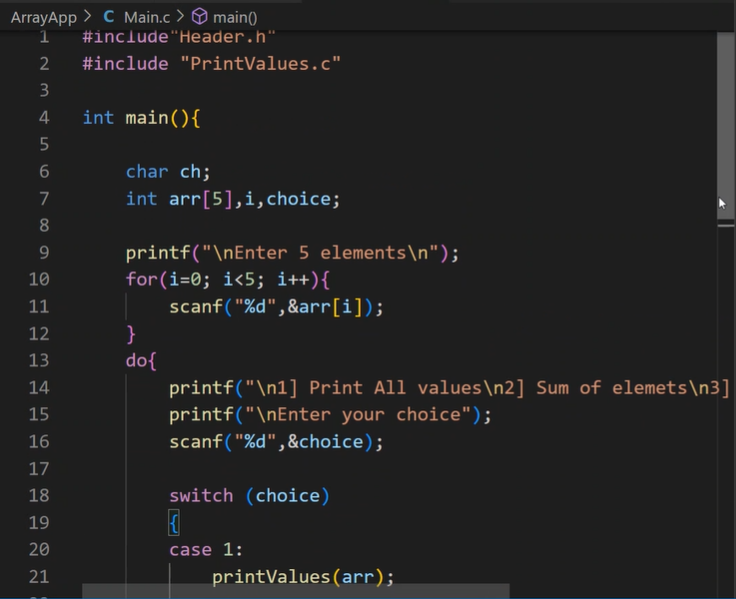


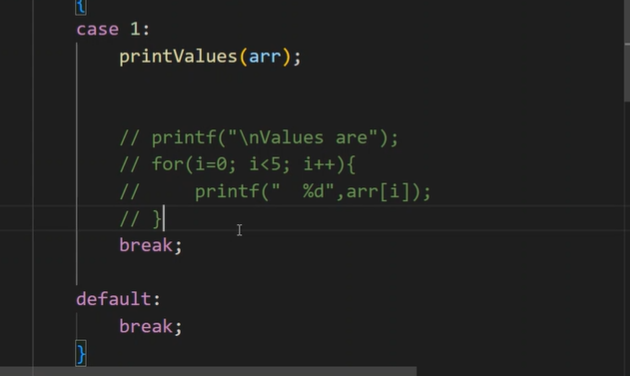
GOD KNOWS WHAT

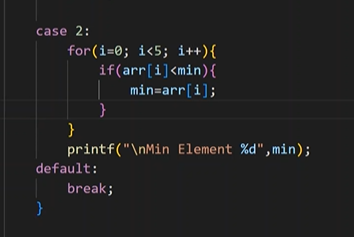


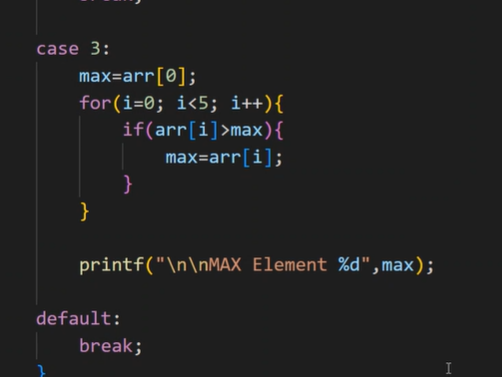
(only print values is done.sum of elements and max elements function is pending)







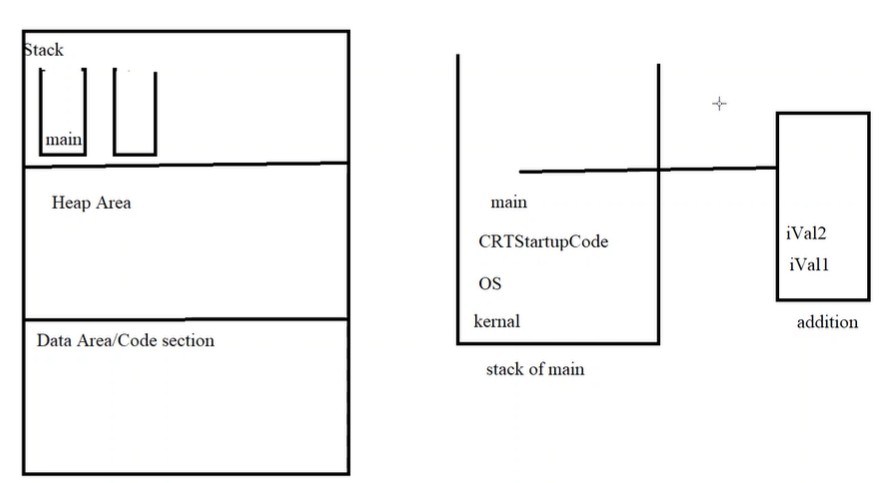


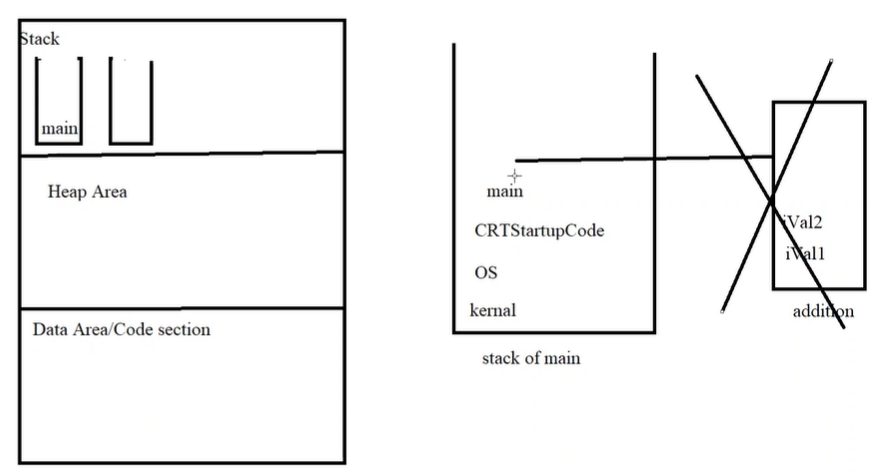


DYNAMIC MEMORY ALLOCATION:

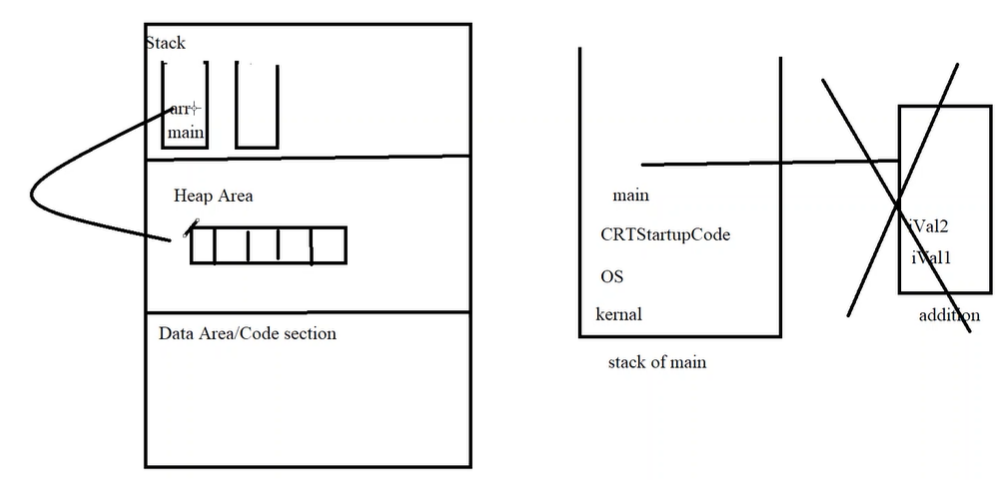
Malloc,calloc

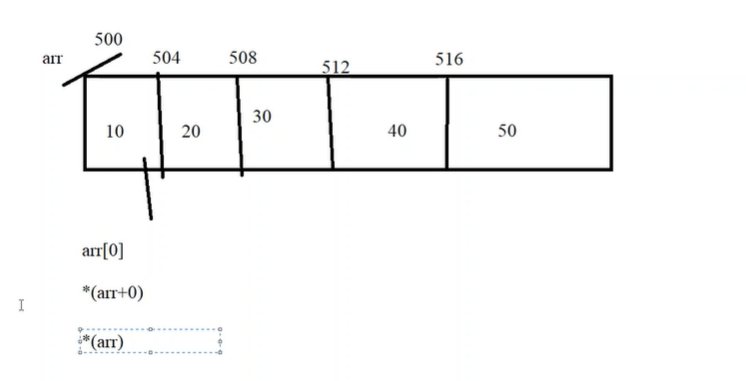
Free(arr)- memory is allocating to the heap location it wont automatically free so we need to manuallyfree the space



After addition: it will automatically free up space

We need to clear heap memory area manually that’s why free(arr) function





CODE:

#include<stdio.h>

#include<malloc.h>

int main(){

    int \*arr,ele,i;

    printf("\nEnter how many element you want to enter\n");

    scanf("%d",&ele);

    arr=(int\*)malloc(sizeof(int)\*ele);

    printf("\nEnter %d elements",ele);

    for(i=0;i<ele;i++){

        scanf("%d",&(\*(arr+i)));

    }

    printf("\nAddresses of all elements\n");

     for(i=0;i<ele;i++){

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

    }

    printf("\narr %u",arr);

    printf("   \*arr %u",\*arr);

    printf("   &arr %u",&arr);

    printf("   \*(arr) %u",\*(arr));

    printf("\narr+1 %u",arr+1);

    printf("    \*arr+1 %u",\*arr+1);

    printf("   &arr+1 %u",&arr+1);

    printf("    \*(arr+1) %u",\*(arr+1));

    printf("\nValues are \n");

     for(i=0;i<ele;i++){

        // printf("  %d",arr[i]);

        printf("  %d",\*(arr+i));

    }

    free(arr);

    return 0;

}

OUTPUT:

