|  |  |  |
| --- | --- | --- |
| Sl. No: | Pg. No: | Type of C Program |
|  |  | **Pointer to Pointer** |
| 1 | 1 | To print value and address of a variable. |
| 2 | 2 | To print value and address of a variable. |
|  |  | **Pointer to Array** |
| 1 | 3 | To print array elements using pointer. |
| 2 | 4 | To print array elements using pointer (arr[x]). |
| 3 | 5 | To print array elements using pointer (x[arr]). |
| 4 | 6 | To print addresses of array elements using pointer (arr+x) |
| 5 | 7 | To print array elements using pointer (\*(arr+x)). |
| 6 | 8 | To print first element in an array using pointer (\*arr). |
|  |  | **Pointer to Multi-Dimensional Array** |
| 1 | 9 | To print the matrix using pointer. |
| 2 | 10-11 | To read the matrix elements and print the matrix using pointer. |
|  |  | **Pointer to Function** |
| 1 | 12 | Sum of two elements using pointer to function. |
| 2 | 13 | Subtraction of two elements using pointer to function. |
|  |  | **Pointer to Arguments** |
| 1 | 14 | Swapping of two elements using pointer to arguments. |
| 2 | 15 | Sum of two elements using pointer to arguments. |
|  |  | **Pointer to Return Types** |
| 1 | 16 | To find the largest number using pointer. |
| 2 | 17 | To print value and address of a variable. |
|  |  | **Pyramid Questions** |
|  |  | **Challenging Questions** |
| 1 |  | To find the maximum number between two numbers using a pointer. |
| 2 |  | To store n elements in an array and print the elements using pointer. |
| 3 |  | To swap elements using call by reference. |
| 4 |  | To find the factorial of a given number using pointers. |
| 5 |  | To sort an array using Pointer. |
| 6 |  | To compute the sum of all elements in an array using pointers. |
| 7 |  | To print all the alphabets using a pointer. |

Pointer to Pointer

To print address and value

#include<stdio.h>

int main()

{

int x=3, \*ptr, \*\*pptr;

ptr=&x;

pptr=&ptr;

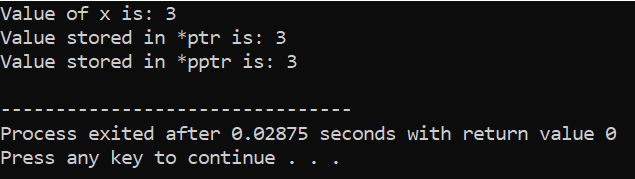
printf("Value of x is: %d\n",x);

printf("Value stored in \*ptr is: %d\n",\*ptr);

printf("Value stored in \*pptr is: %d\n",\*\*pptr);

return 0;

}



To print address and value

#include <stdio.h>

int main() {

int a = 3, \*ptr, \*\*pptr;

ptr=&a;

pptr=&ptr;

printf("Address of a = %p\n", &a);

printf("Address of ptr = %p\n", &ptr);

printf("Address of pptr = %p\n\n", &pptr);

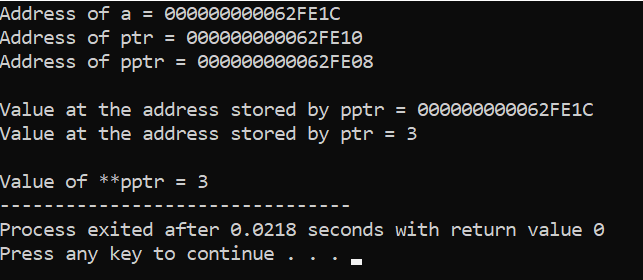
printf("Value at the address stored by pptr = %p\n", \*pptr); //here address will be printed

printf("Value at the address stored by ptr = %d\n\n", \*ptr);

printf("Value of \*\*pptr = %d", \*\*pptr);

return 0;

}



Pointer to Array

To print array elements using pointer

#include<stdio.h>

int main()

{

int x, \*p;

int arr[5]={1,2,2,3,7};

p=&arr[0];

printf("Elements in the array are:\n");

for(x=0;x<5;x++)

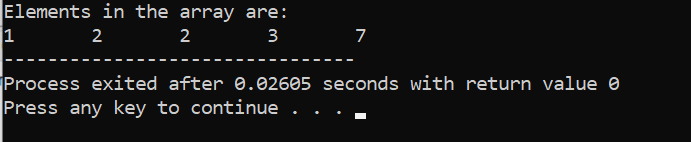
{ printf("%d\t",\*p);

p++;

}

return 0;

}



To print array elements using pointer (arr[x] in printf)

#include<stdio.h>

int main()

{

int x, \*p;

int arr[5]={1,2,2,3,7};

p=arr;

printf("Elements in the array are:\n");

for(x=0;x<5;x++)

{

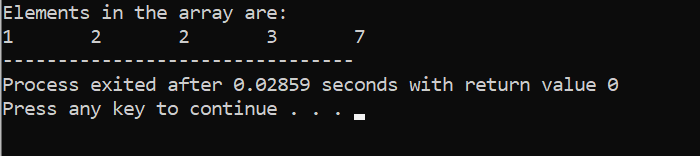
printf("%d\t",**arr[x]**);

p++;

}

return 0;

}



To print array elements using pointer (x[arr] in printf)

#include<stdio.h>

int main()

{

int x, \*p;

int arr[5]={1,2,2,3,7};

p=arr;

printf("Elements in the array are:\n");

for(x=0;x<5;x++)

{

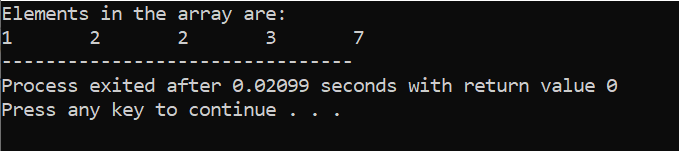
printf("%d\t",**x[arr]**);

p++;

}

return 0;

}



To print addresses of array elements using pointer

(arr+x in printf)

#include<stdio.h>

int main()

{

int x, \*p;

int arr[5]={1,2,2,3,7};

p=arr;

printf("Addresses of the array are:\n");

for(x=0;x<5;x++)

{

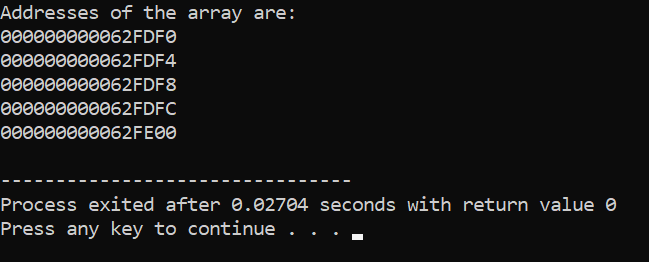
printf("%p\n", **arr+x**);

p++;

}

return 0;

}



To print array elements using pointer (\*(arr+x) in printf)

#include<stdio.h>

int main()

{

int x, \*p;

int arr[5]={1,2,2,3,7};

p=arr;

printf("Elements in the array are:\n");

for(x=0;x<5;x++)

{

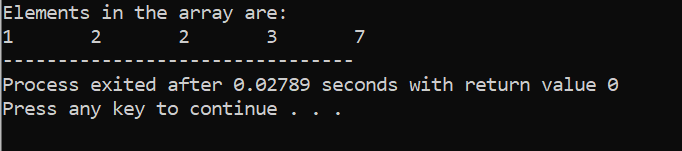
printf("%d\t", **\*(arr+x)**);

p++;

}

return 0;

}



To print first element of array using pointer (\*arr in printf)

#include<stdio.h>

int main()

{

int x, \*p;

int arr[5]={1,2,2,3,7};

p=arr;

for(x=0;x<5;x++)

{

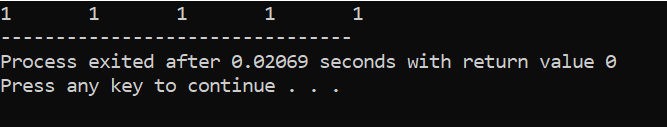
printf("%d\t", **\*arr**);

p++;

}

return 0;

}



Pointer to Multi-Dimensional Array

To print matrix using pointer

#include<stdio.h>

int main()

{

int x,y;

int matrix[3][3]={

{1,2,2},

{4,1,7},

{3,5,4}

};

printf("Matrix you entered is:\n");

for(x=0;x<3;x++)

{

for(y=0;y<3;y++)

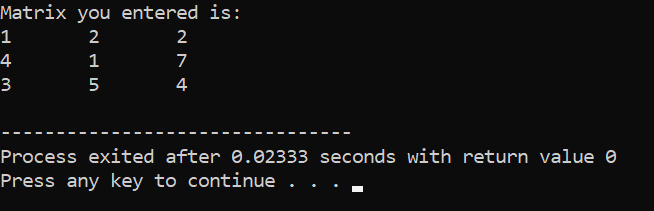
{ printf("%d\t",\*(\*(matrix+x)+y)); }

printf("\n");

}

return 0;

}



To read the matrix elements and print matrix using pointer

#include<stdio.h>

int main()

{

int x,y,matrix[3][3];

printf("Enter the values in the matrix\n");

for(x=0;x<3;x++)

{

for(y=0;y<3;y++)

{

printf("Element-%d,%d",x,y);

scanf("%d",&\*(\*(matrix+x)+y));

}

}

printf("Matrix you entered is:\n");

for(x=0;x<3;x++)

{

for(y=0;y<3;y++)

{

printf("%d\t",\*(\*(matrix+x)+y));

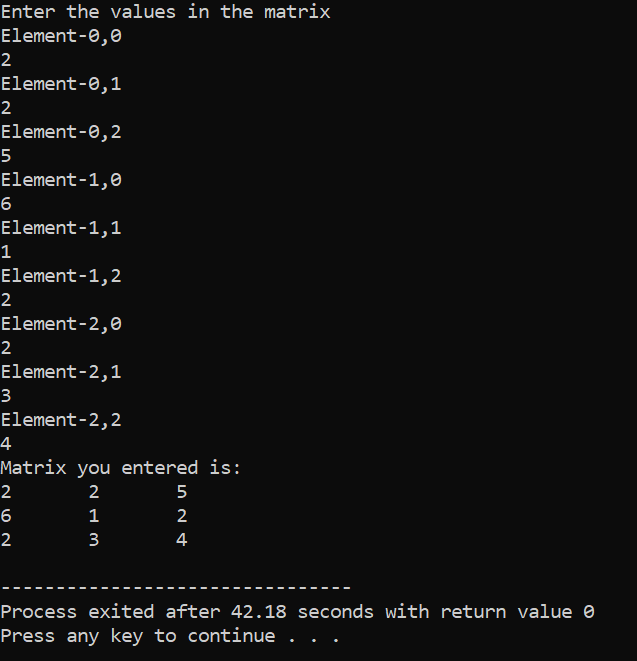
}

printf("\n");

}

return 0;

}



Pointer to Function

Sum of two elements using pointer

#include <stdio.h>

int sum(int x, int y)

{

return x+y;

}

int main()

{

int (\*ptr)(int, int);

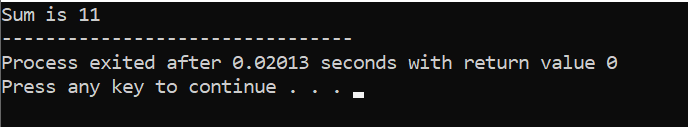
ptr = sum;

int s = ptr(6, 5);

printf("Sum is %d", s);

return 0;

}



Subtraction of two elements using pointer

#include <stdio.h>

int sub(int x, int y)

{

return x-y;

}

int main()

{

int (\*ptr)(int, int);

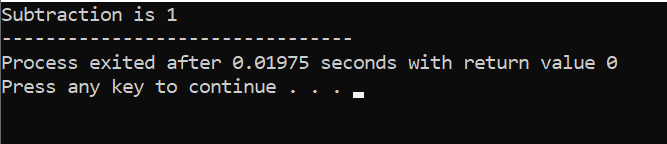
ptr = sub;

int s = ptr(6, 5);

printf("Subtraction is %d", s);

return 0;

}



Pointer to Arguments

Swapping of two elements using pointer

#include<stdio.h>

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

{ int c;

c=\*a;

\*a=\*b;

\*b=c; }

int main()

{

int x=2,y=3;

printf("Before Swapping\n");

printf("x=%d\n",x);

printf("y=%d\n",y);

swap(&x,&y);

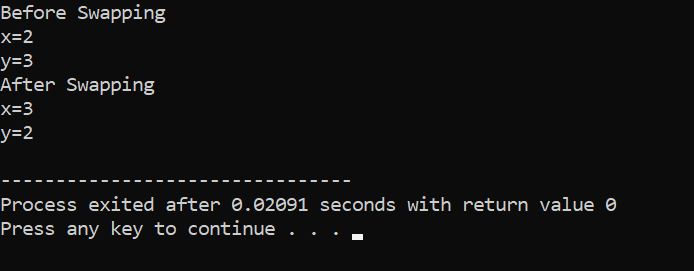
printf("After Swapping\n");

printf("x=%d\n",x);

printf("y=%d\n",y);

return 0;

}



Addition of two elements using pointer

#include<stdio.h>

int add(int \*a, int \*b)

{

int c;

c=\*a+\*b;

return c;

}

int main()

{

int x,y,z;

printf("Enter two integers to sum\n");

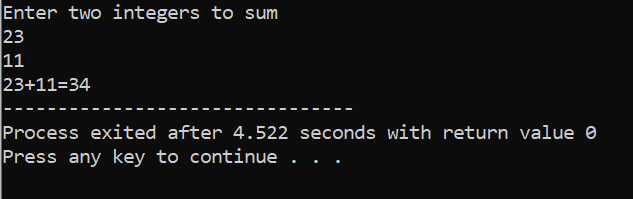
scanf("%d%d",&x,&y);

z=add(&x,&y);

printf("%d+%d=%d",x,y,z);

return 0;

}



Pointer to Return Types

To find the largest number using pointer

#include<stdio.h>

int \*largest(int \*a, int \*b)

{

if(\*a<\*b)

return b;

else

return a;

}

int main()

{

int x,y,\*z;

printf("Enter any two integers to find the largest\n");

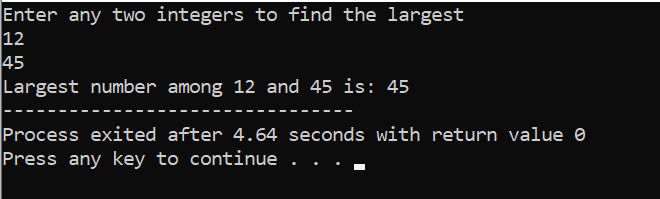
scanf("%d%d",&x,&y);

z=largest(&x,&y);

printf("Largest number among %d and %d is: %d",x,y,\*z);

return 0;

}



To print value and address of a variable using pointer

#include<stdio.h>

int \*print()

{

static int a=5;

return &a;

}

int main()

{

int \*n;

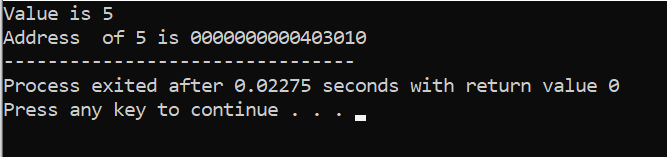
n=print();

printf("Value is %d\n",\*n);

printf("Address of %d is %p",\*n,n);

return 0;

}



**Pyramid Questions & Challenging Questions**

#include<stdio.h>

int main()

{

int x,y,n;

printf("Enter the no. of rows to print half pyramid\n");

scanf("%d",&n);

for(x=0;x<n;x++)

{

for(y=0;y<=x;y++)

{

printf("\*");

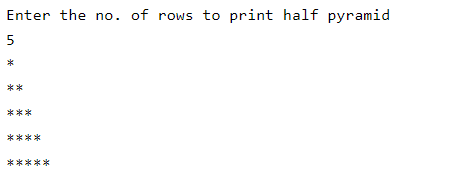
}

printf("\n");

}

return 0;

}



#include<stdio.h>

int main()

{

int x,y,n;

printf("Enter the no. of rows to print half pyramid\n");

scanf("%d",&n);

for(x=n-1;x>=0;x--)

{

for(y=0;y<=x;y++)

{

printf("\*");

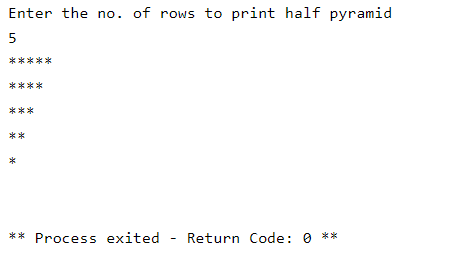
}

printf("\n");

}

return 0;

}



#include<stdio.h>

int main()

{

int x,y,n;

printf("Enter the no. of rows to print half pyramid\n");

scanf("%d",&n);

for(x=1;x<=n;x++)

{

for(y=1;y<=x;y++)

{

printf("%d",y);

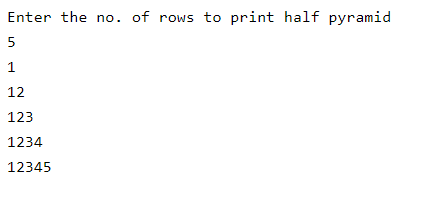
}

printf("\n");

}

return 0;

}



#include<stdio.h>

int main()

{

int x,y,n;

printf("Enter the no. of rows to print half pyramid\n");

scanf("%d",&n);

for(x=n;x>=1;x--)

{

for(y=1;y<=x;y++)

{

printf("%d",y);

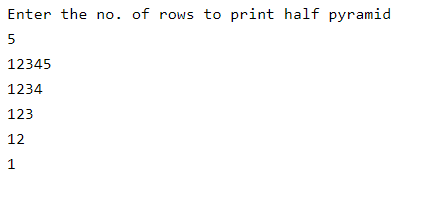
}

printf("\n");

}

return 0;

}



#include<stdio.h>

int main()

{

int x,y,\*ptr1=&x,\*ptr2=&y;

printf("Enter two numbers");

scanf("%d%d",&x,&y);

if(x>y)

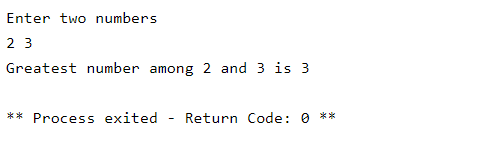
printf("Greatest number among %d and %d is %d",\*ptr1,\*ptr2,\*ptr1);

else

printf("Greatest number among %d and %d is %d",\*ptr1,\*ptr2,\*ptr2);

return 0;

}



#include<stdio.h>

int main()

{

int x,n,arr[100];

printf("Enter the array limit\n");

scanf("%d",&n);

printf("Enter those %d elements\n",n);

for(x=0;x<n;x++)

{

printf("Element-%d",x);

scanf("%d",arr+x);

}

printf("Elements in the array are:\n");

for(x=0;x<n;x++)

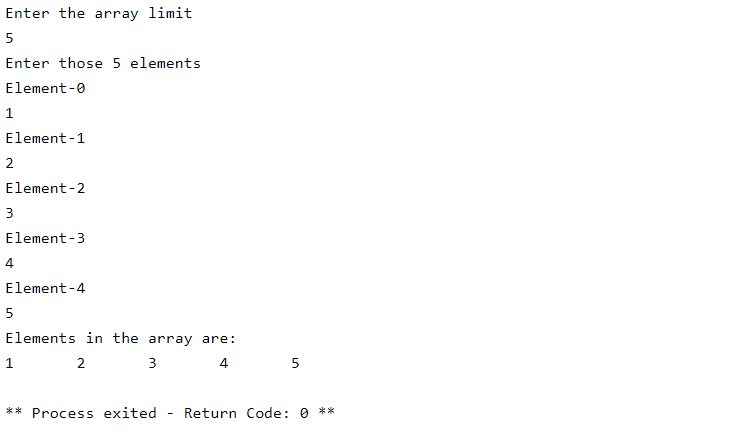
{

printf("%d\t",\*(arr+x));

}

return 0;

}



#include<stdio.h>

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

{ int c;

c=\*a;

\*a=\*b;

\*b=c; }

int main()

{

int x=7,y=4;

printf("Before Swapping\n");

printf("x=%d\n",x);

printf("y=%d\n",y);

swap(&x,&y);

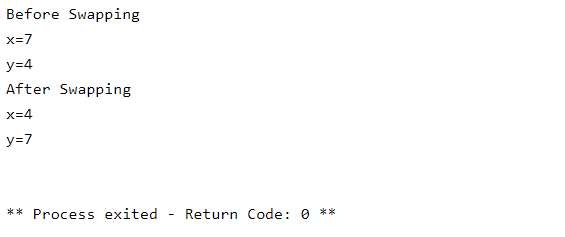
printf("After Swapping\n");

printf("x=%d\n",x);

printf("y=%d\n",y);

return 0;

}



#include<stdio.h>

int factorial(int a,int \*b)

{

int c;

\*b=1;

for(c=1;c<=a;c++)

{

\*b=\*b\*c;

}

}

int main()

{

int x,y,z;

printf("Enter the value\n");

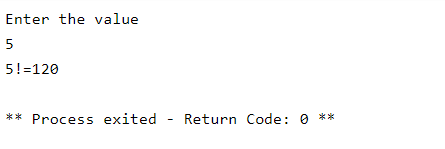
scanf("%d",&x);

factorial(x,&y);

printf("%d!=%d",x,y);

return 0;

}



#include<stdio.h>

int main()

{

int x,y,z,a;

int sort[100];

printf("Number of values to be considering");

scanf("%d",&x);

printf("Enter those %d numbers\n",x);

for(y=0;y<x;y++)

{

printf("Element -%d\n",y);

scanf("%d",sort+y);

}

printf("The elements you have entered:\n");

for(y=0;y<x;y++)

{

printf("%d\t",\*(sort+y));

}

for(y=0;y<x;y++)

{

for(z=y+1;z<x;z++)

{

if(\*(sort+y)>\*(sort+z))

{

a=\*(sort+y);

\*(sort+y)=\*(sort+z);

\*(sort+z)=a;

}

}

}

printf("\nThe elements after arranging in ascending order is:\n");

for(y=0;y<x;y++)

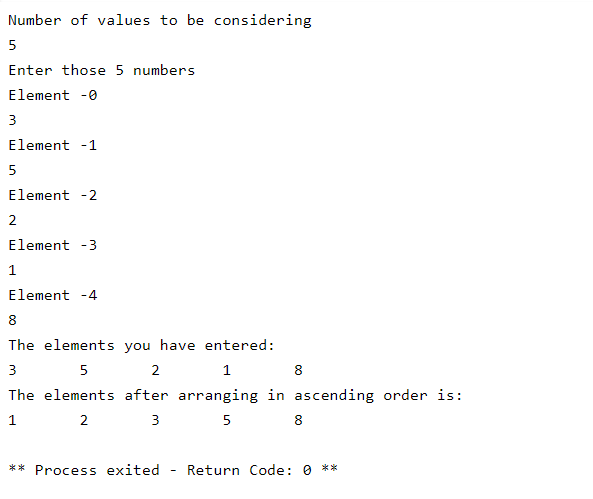
{

printf("%d\t",\*(sort+y));

}

return 0;

}



#include<stdio.h>

int main()

{

int x,a=0,n,arr[100];

printf("Enter the array limit\n");

scanf("%d",&n);

printf("Enter those %d elements\n",n);

for(x=0;x<n;x++)

{

printf("Element-%d",x);

scanf("%d",arr+x);

}

for(x=0;x<n;x++)

{

a+=\*(arr+x);

}

printf("Sum of all the elements in the array is: %d",a);

return 0;

}

