1. CC with DFT-IDFT method

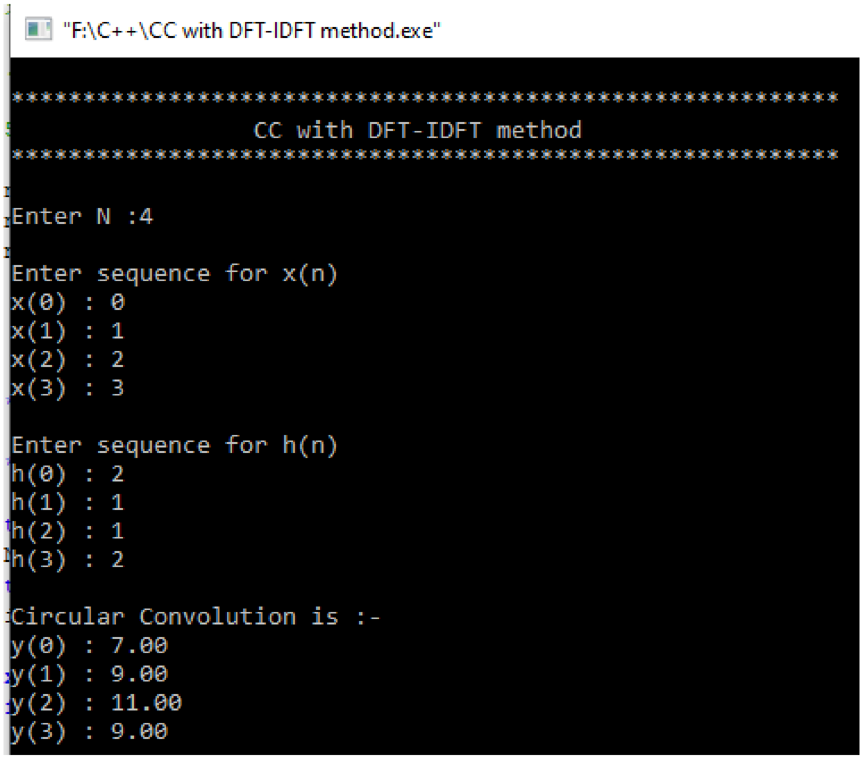
Code:

```
#include<stdio.h>
#include<math.h>
#include<complex.h>

#define PI 3.1415

float static x_input[100],x_real[100],x_img[100];
float static h_input[100],h_real[100],h_img[100];
float static y_input[100],y_real[100],y_img[100];

int main(void)
{
   int i,j,N;
   printf("\n***************************************************************\n");
   printf("\t\t CC with DFT-IDFT method");
   printf("\n***************************************************************\n");

   printf("\nEnter N :");
   scanf("%d",&N);
   printf("\nEnter sequence for x(n)\n");
   for(i=0;i<N;i++)
   {
      printf("x(%d) : ",i);
      scanf("%f",&x_input[i]);
   }

   printf("\nEnter sequence for h(n)\n");
   for(i=0;i<N;i++)
   {
      printf("h(%d) : ",i);
      scanf("%f",&h_input[i]);
   }

   for(i=0;i<N;i++)                //DFT for x(n)
   {
      x_real[i]=x_img[i]=0.0;
      for(j=0;j<N;j++)
      {
         x_real[i]=x_real[i]+x_input[j]*cos((2*PI*i*j)/N);
         x_img[i]=x_img[i]+x_input[j]*sin((2*PI*i*j)/N);
      }
      x_img[i]=x_img[i]*(-1.0);
   }
   for(i=0;i<N;i++)                //DFT for h(n)
   {
      h_real[i]=h_img[i]=0.0;
```

```
        for(j=0;j<N;j++)
        {
            h_real[i]=h_real[i]+h_input[j]*cos((2*PI*i*j)/N);
            h_img[i]=h_img[i]+h_input[j]*sin((2*PI*i*j)/N);
        }
        h_img[i]=h_img[i]*(-1.0);
    }

    for(i=0;i<N;i++)
    {
        y_real[i]=x_real[i]*h_real[i]-x_img[i]*h_img[i];
        y_img[i]=x_real[i]*h_img[i]+x_img[i]*h_real[i];
    }

    for(j=0;j<N;j++)            // IDFT
    {
        for(i=0;i<N;i++)
        {
            y_input[j]=y_input[j]+y_real[i]*cos((2*PI*j*i)/N)-y_img[i]*sin((2*PI*i*j)/N);
        }
        y_input[j]=y_input[j]/N;
    }


    printf("\nCircular Convolution is :-\n");
    for(i=0;i<N;i++)
    {
        printf("y(%d) : %.2f\n",i,y_input[i]);
    }

}
```

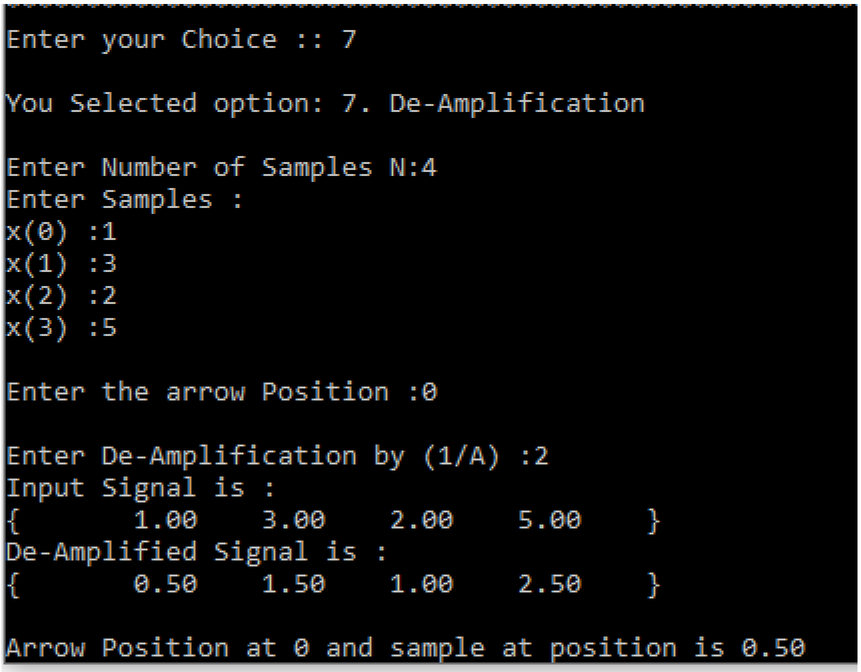
Output:

CC with DFT-IDFT method



"F:\C++\CC with DFT-IDFT method.exe"

```
**********************************************************
                 CC with DFT-IDFT method
**********************************************************

Enter N :4

Enter sequence for x(n)
x(0) : 0
x(1) : 1
x(2) : 2
x(3) : 3

Enter sequence for h(n)
h(0) : 2
h(1) : 1
h(2) : 1
h(3) : 2

Circular Convolution is :-
y(0) : 7.00
y(1) : 9.00
y(2) : 11.00
y(3) : 9.00
```



```
Enter your Choice :: 7

You Selected option: 7. De-Amplification

Enter Number of Samples N:4
Enter Samples :
x(0) :1
x(1) :3
x(2) :2
x(3) :5

Enter the arrow Position :0

Enter De-Amplification by (1/A) :2
Input Signal is :
{       1.00    3.00    2.00    5.00    }
De-Amplified Signal is :
{       0.50    1.50    1.00    2.50    }

Arrow Position at 0 and sample at position is 0.50
```