**/\*Write a program to perform Linear convolution using TMS320C6748 DSP Platform\*/  
  
#include<stdio.h>  
int y[7];  
void main()  
{  
    int m=4; //Length of input sample sequence  
    int n=4; // Length of impulse response Co-efficient  
    int i,j;  
    int x[7]={1,2,3,4};//Input signal sample  
    int h[7]={4,3,2,1};//Impulse Response Co-efficient  
            for(i=0;i<m+n-1;i++)  
            {  
                y[i]=0;  
                for(j=0;j<=i;j++)  
                {  
                    y[i]+=x[j]\*h[i-j];  
                }  
  
            }**

**printf("Linear Convolution output\n:");  
            for(i=0;i<m+n-1;i++)  
            {  
                printf("%d\n",y[i]);  
            }  
}**

**/\*Write a program to perform DFT using TMS320C6748 DSP Platform\*/  
  
#include<stdio.h>  
#include<math.h>  
void dft(float \*x,short k);  
#define N 3  
float pi=3.1416;  
float x[N]={0.25,0.25,0.25};  
void dft(float \*x,short k) //dft function  
{  
    float sumRe=0, sumIm=0; // init real/imag components  
    float cs=0,sn=0; //init cosine/sine components  
    int i=0;  
    for (i=0;i<N;i++)// for N point DFT  
    {  
        cs=cos(2\*pi\*(k)\*i/N);// real component  
        sn=sin(2\*pi\*(k)\*i/N);//imaginary component  
        sumRe=sumRe+x[i]\*cs;//sum of real components  
        sumIm=sumIm-x[i]\*sn;//sum of imaginary components  
    }  
    printf("%f\n",sumRe);  
}  
void main()  
{  
    int j;  
    for(j=0;j<N;j++)  
    {  
        dft(x,j);// call DFT function  
    }  
}**