Overlapsave.c

#include<stdio.h>

int cir\_out[20][10];

void circonv(int x[],int h[], int len, int arr\_flag){

int i,k,modval;

int maxlen = len;

int y[maxlen];

for(i=0;i<maxlen;i++)

{

y[i]=0;

for(k=0;k<maxlen;k++)

{

if((i-k)<0)

{

modval = maxlen+(i-k);

y[i] += (x[k]\*h[modval]);

}

else

{

y[i] += (x[k]\*h[(i-k)]);

}

cir\_out[arr\_flag][i] = y[i];

}

}

for(i=0;i<maxlen;i++)

{

printf("%d\t",cir\_out[arr\_flag][i]);

}

}

int main(){

int h[20], x[20];

int lx,lh,i,j=0;

printf("Enter length of x[n]\n");

scanf("%d",&lx);

printf("Enter length of h[n]\n");

scanf("%d",&lh);

printf("Enter elements for x[n]\n");

for(i=0;i<lx;i++)

{

scanf("%d",&x[i]);

}

printf("Enter elements for h[n]\n");

for(i=0;i<lh;i++)

{

scanf("%d",&h[i]);

}

int l = 4 , m = lh, n = l+m-1;

int temp[lx+lh-1], set\_arr[n], tempFlag=0, overlap\_arr\_counter=3, count = 0;

if(lh < n)

{

for(i=lh;i<n;i++)

h[i] = 0;

}

for(i=0 ; i<(lx+lh-1) ; i++)

{

if(i < m-1)

{

temp[i] = 0;

}

else

{

temp[i] = x[j];

j++;

}

}

while(overlap\_arr\_counter != -1)

{

j=0;

for(i=0+(l\*tempFlag) ; i<n+((l+1)\*tempFlag) ; i++)

{

if(i < lx+lh-1)

set\_arr[j] = temp[i];

else

set\_arr[j] = 0;

printf("%d\n",i);

j++;

}

circonv(set\_arr, h, n, tempFlag);

tempFlag++;

if(i >= lx+lh+1)

overlap\_arr\_counter = -1;

}

printf("\n");

for(i=0 ; i<4; i++)

for(j=l; j<n; j++)

printf("%d\t", cir\_out[i][j]);

printf("\n");

}

\*/OUTPUT

Enter length of x[n]

13

Enter length of h[n]

4

Enter elements for x[n]

1 2 -1 3 -2 1 0 3 4 -2 3 1 4

Enter elements for h[n]

1 -1 0 1

Linear convolution by overlapsave

1 1 -3 5 -3 2 2 1 2 -6 8 2 -2 2 1 1 \*/