```
// This is the main project file for VC++ application project

#include "stdafx.h"
#using <mscorlib.dll>
#include "stdio.h"
using namespace System;

double DigitalFilter(double x);

int _tmain()
{
      double x[] = {1.1,2.1,3.4,5.6,7.8,3.4,2.2,1.1}; // the input vector
      double y[8];
      int i;
      char dummy;

      for (i=0;i<8;i++){
            y[i] = DigitalFilter(x[i]);
            Console::WriteLine(y[i]);  // Note: processes in real time!
      }
      printf("Press y and <enter>");
      scanf("%ls",&dummy);
      return 0;
}



double DigitalFilter(double x)
/*************************************************************************/
/* Inputs:                                                               */
/*   x        float       The input to the filter                        */
/*   This is a second order example  H(z) = 8 + 6z^-1 + 2z^-2            */
/*                                          ---------------------        */
/*                                          1 - z^-1 + 0.5z^-2           */
/*                                                                       */
/* Note: length of zden must always be one less than the length of znum  */
/*       zero pad the end of zden if necessary (e.g. for an FIR filter)  */
/*************************************************************************/

{
      const int ORDER=2;                // IIR filter_order
      double sumnum=0, sumden=0;
      static double delay[] = {0,0,0};  // as many as ORDER+1
      static double znum[] = {2, 6, 8}; //znum[2], etc. backwards
      static double zden[] = {0.5, -1}; //backwards, but zden[0]=1 so not entered
      for (int i=0;i<ORDER;i++){
            delay[i] = delay[i+1];
            sumden += delay[i]*zden[i];
            sumnum += delay[i]*znum[i];
      }
      delay[ORDER] = x-sumden;
      sumnum += delay[ORDER]*znum[ORDER];
      return sumnum;
}
```