1 Consider the following program:

1 #include <stdio.h>

2 #define DATALEN 15

3 #define FILTERTAPS 5

4 double x[DATALEN] = { 128.0, 130.0, 180.0, 140.0, 120.0,

5 110.0, 107.0, 103.5, 102.0, 90.0,

6 84.0, 70.0, 30.0, 77.3, 95.7 };

7 const double h[FILTERTAPS]={0.125,-0.25,0.5,-0.25,0.125};

8 double y[DATALEN]; // result;

9 int main(void)

10 {int i,n;

11 for(i=0;i<DATALEN;++i)

12 {y[i]=0;

13 for(n=0;n<FILTERTAPS;++n)

14 if ((i-n)>=0) y[i]+=h[n]\*x[i-n];

15 }

16 for(i=0;i<DATALEN;++i) printf("%.2f ",y[i]);

17 return 0;

18 }

Perform at least the following optimizations:

Removal of the if in the innermost loop (line 14)

Loop unrolling (line 13)

Constant propagation

Floating-point to fixed-point conversion

Avoidance of all accesses to arrays

Please provide the optimized version of the program after each of the transformations

and do also check for consistent results!

1 #include <stdio.h>

2 #define DATALEN 15

3 #define FILTERTAPS 5

4 double x[DATALEN] = { 128.0, 130.0, 180.0, 140.0, 120.0,

5 110.0, 107.0, 103.5, 102.0, 90.0,

6 84.0, 70.0, 30.0, 77.3, 95.7 };

7 const double h[FILTERTAPS]={0.125,-0.25,0.5,-0.25,0.125};

8 double y[DATALEN]; // result;

9 int main(void)

10 {int i,n;

11 for(i=0;i<DATALEN;++i)

12 {y[i]=0;

13 for(n=0;n≤FILTERTAPS-1 && n≤i ;++n)

14 y[i]+=h[n]\*x[i-n];

15 }

16 for(i=0;i<DATALEN;++i) printf("%.2f ",y[i]);

17 return 0;

18 }