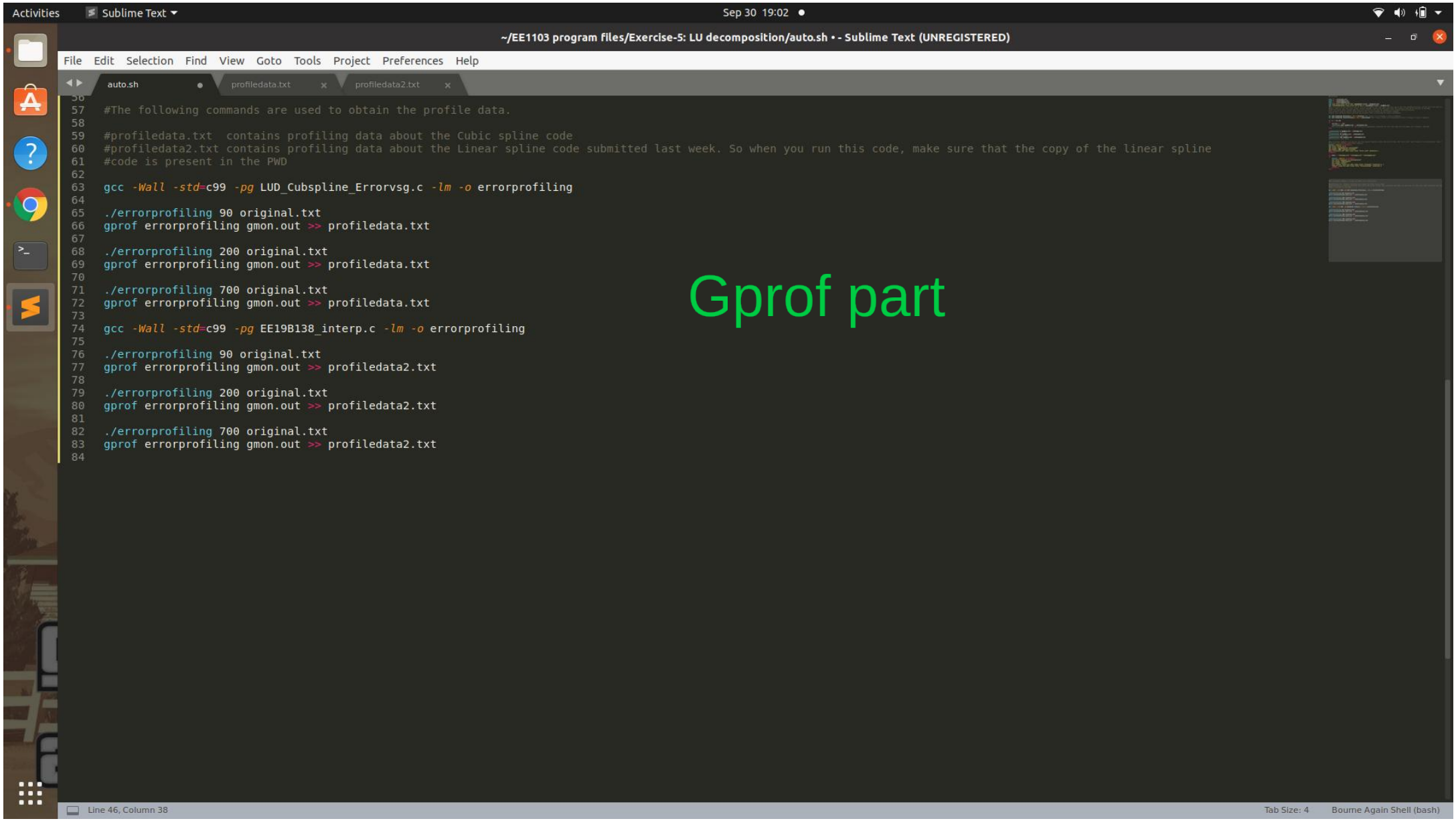
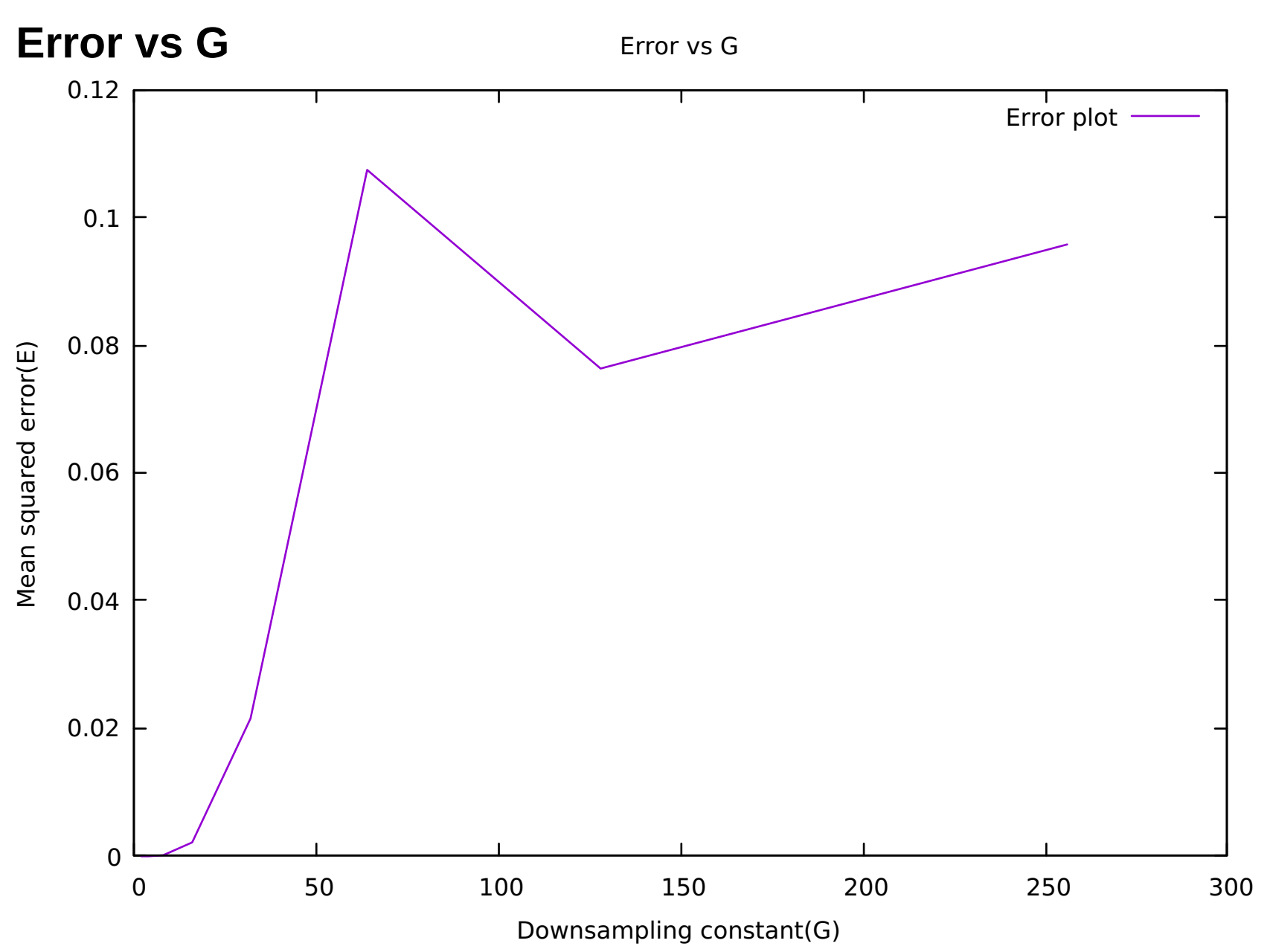
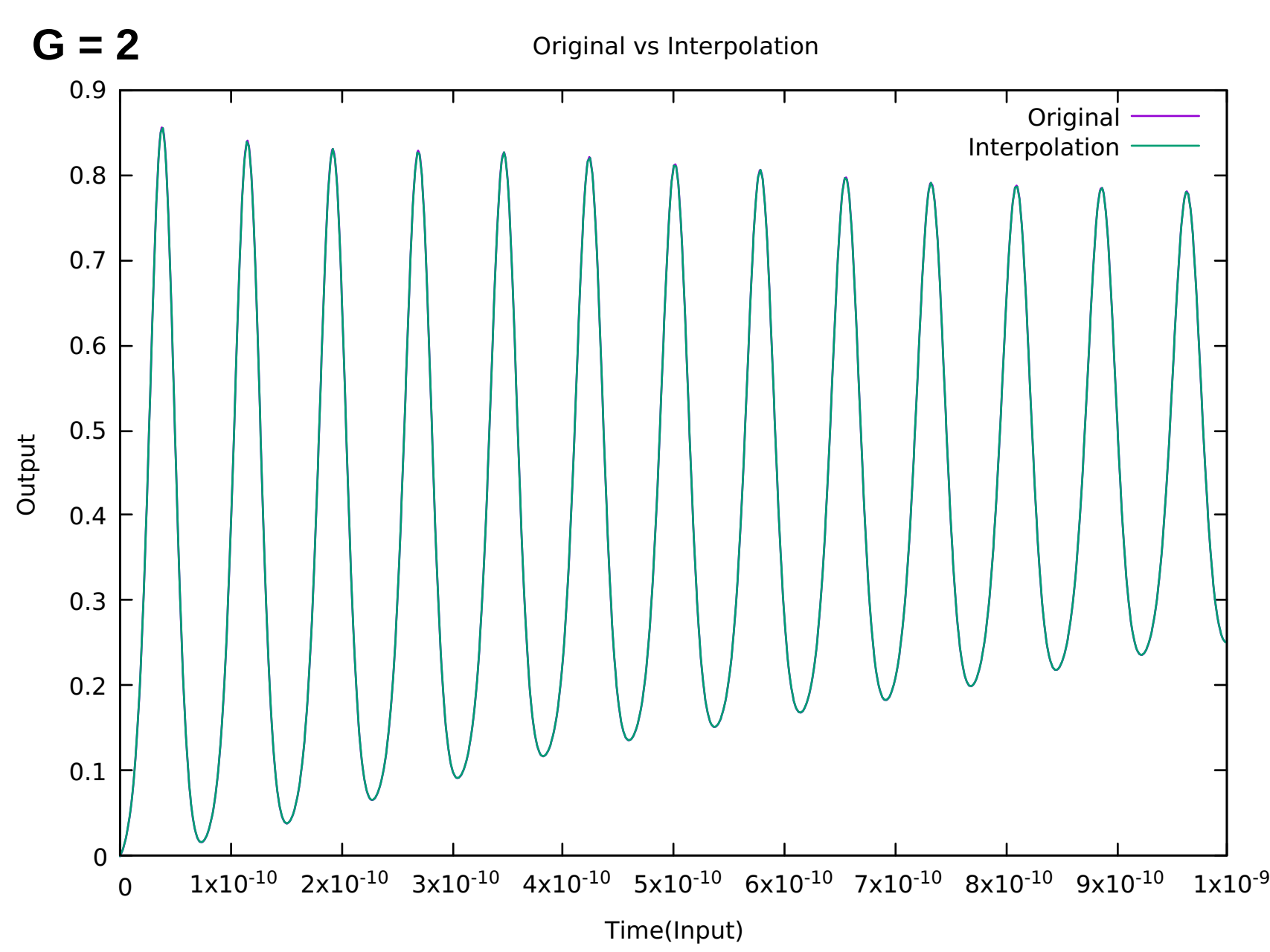
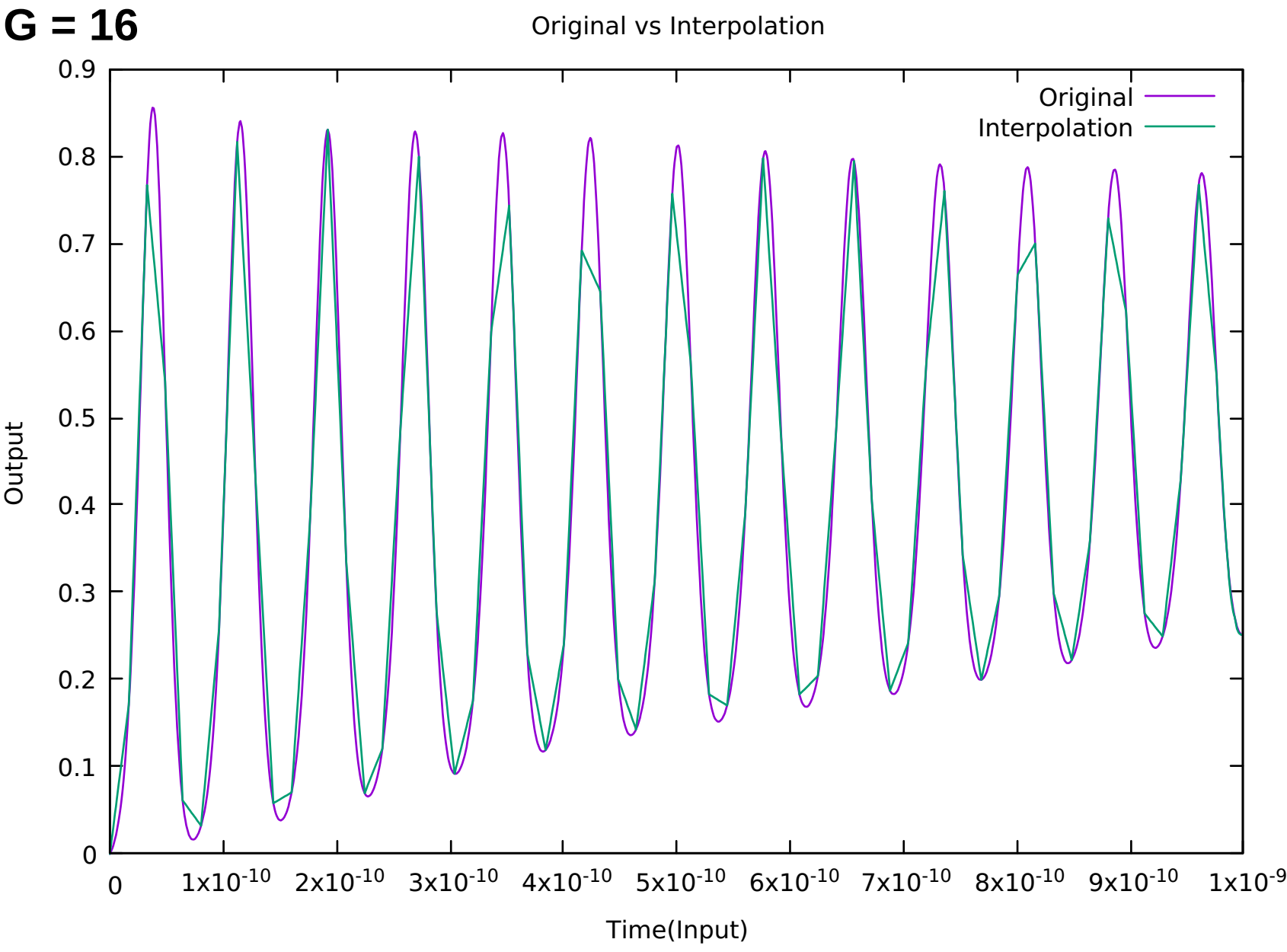
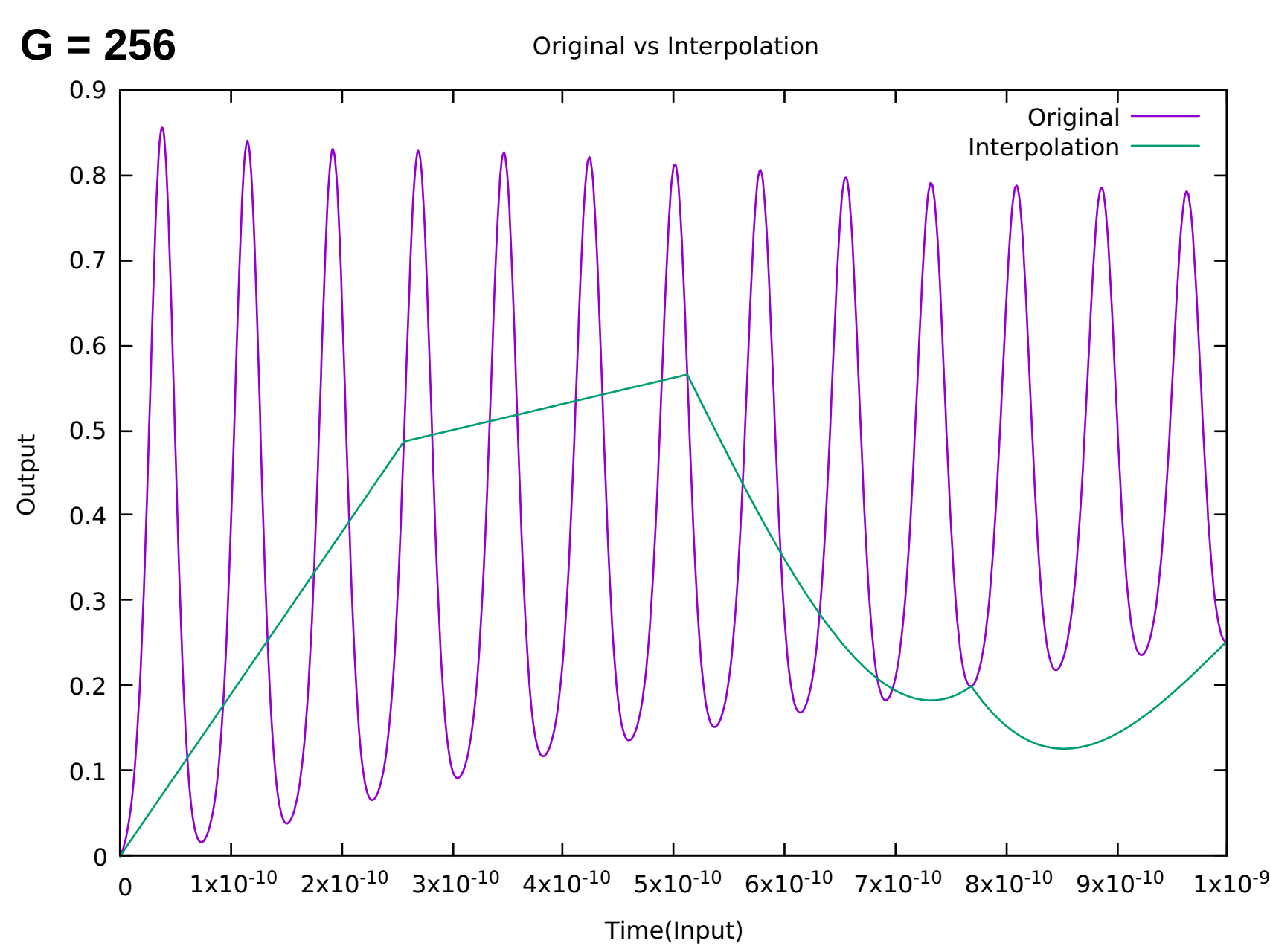

Activities
Sublime Text
Sep 30 19:01
~/EE1103 program files/Exercise-5: LU decomposition/auto.sh • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
auto.sh
profiledata.txt
profiledata2.txt
#!/bin/bash
echo "" > errorplot.txt
echo "" > profiledata.txt
echo "" > profiledata2.txt
awk '{$1 = $1; print $1"\t"$2}' assignment_2.txt > original.txt
awk '{if(NR<1001){$1 = $1; print $1"\t"$2} }' assignment_2.txt > newdata.txt
#The file path of the datafile must be free of any spaces (In this case, the file was in the same working directory. So the full file path isn't needed.)
#Hence, when you run this bash script, please provide a space-free file path (or) just create a copy of the dataset in the PWD.
All the other files created when this bash script is run, will be created in the present working diretory.
So we don't need to specify their paths in the ./a.out command or the gnuplot command.
Also, only the first 1000 rows from the original file is extracted for space constraints.
gcc LUD_Cubspline_Errorvsg.c -lm -o errorvsg #The "Error vs G" printing C code is compiled
gcc LUD_Cubspline_Inoutinterpol.c -lm -o inoutintpol #The "Input, Output and Interpolated output" printing C code is compiled
for i in {1..8}
do
var=$((2 ** i))
./errorvsg $var newdata.txt >> errorplot.txt
#The executable file is ran for different downsampling constants (in this case they are the powers of 2 between 2 and 256)
done
./inoutintpol 2 newdata.txt > interpg2.txt
#Interolation with g=2
./inoutintpol 16 newdata.txt > interpg16.txt
#Interpolation with g=16
./inoutintpol 256 newdata.txt > interpg256.txt
#Interpolation with g=256
#The following commands are used to call the gnuplot commands within the bash script. The 'Error plot' and 'Original vs interpolation' plots for some downsampling
#constants are plotted using these commands
gnuplot -persist <<-EOFMarker
set title 'Error vs G'
set xlabel 'Downsampling constant(G)'
set ylabel 'Mean squared error(E)'
plot 'errorplot.txt' with lines title 'Error plot' linestyle 1
EOFMarker
for FILE in "interpg2.txt" "interpg16.txt" "interpg256.txt"
do
gnuplot -persist <<-EOFMarker
set title 'Original vs Interpolation'
set xlabel 'Time(Input)'
set ylabel 'Output'
plot '$FILE' using 1:2 with lines title 'Original' linestyle 1, \
'$FILE' using 1:3 with lines title 'Interpolation' linestyle 2
EOFMarker
done
Bash script
Gnuplot part
Line 57, Column 1
Tab Size: 4
Bourne Again Shell (bash)


Activities
Sublime Text
Sep 30 19:02
~/EE1103 program files/Exercise-5: LU decomposition/auto.sh • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
auto.sh
profiledata.txt
profiledata2.txt
#The following commands are used to obtain the profile data.
#profiledata.txt contains profiling data about the Cubic spline code
#profiledata2.txt contains profiling data about the Linear spline code submitted last week. So when you run this code, make sure that the copy of the linear spline
#code is present in the PWD
gcc -Wall -std=c99 -pg LUD_Cubspline_Errorvsg.c -lm -o errorprofiling
./errorprofiling 90 original.txt
gprof errorprofiling gmon.out >> profiledata.txt
./errorprofiling 200 original.txt
gprof errorprofiling gmon.out >> profiledata.txt
./errorprofiling 700 original.txt
gprof errorprofiling gmon.out >> profiledata.txt
gcc -Wall -std=c99 -pg EE19B138_interp.c -lm -o errorprofiling
./errorprofiling 90 original.txt
gprof errorprofiling gmon.out >> profiledata2.txt
./errorprofiling 200 original.txt
gprof errorprofiling gmon.out >> profiledata2.txt
./errorprofiling 700 original.txt
gprof errorprofiling gmon.out >> profiledata2.txt
Gprof part
Line 46, Column 38
Tab Size: 4
Bourne Again Shell (bash)


Error vs G
Error vs G
Error plot
0.12
0.1
0.08
0.06
0.04
0.02
0
Mean squared error(E)
0
50
100
150
200
250
300
Downsampling constant(G)


G = 2
Original vs Interpolation
Original
Interpolation
Output
Time(Input)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0
1x10-10
2x10-10
3x10-10
4x10-10
5x10-10
6x10-10
7x10-10
8x10-10
9x10-10
1x10-9


G = 16
Original vs Interpolation
Original
Interpolation
Output
Time(Input)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0
1x10-10
2x10-10
3x10-10
4x10-10
5x10-10
6x10-10
7x10-10
8x10-10
9x10-10
1x10-9


G = 256
Original vs Interpolation
Original
Interpolation
Output
Time(Input)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
0
1x10-10
2x10-10
3x10-10
4x10-10
5x10-10
6x10-10
7x10-10
8x10-10
9x10-10
1x10-9

```
Flat profile:

Each sample counts as 0.01 seconds.
  %   cumulative   self              self     total
 time   seconds   seconds    calls  Ts/call  Ts/call  name
100.17      0.48     0.48                             main

 %         the percentage of the total running time of the
time       program used by this function.

cumulative a running sum of the number of seconds accounted
 seconds   for by this function and those listed above it.

 self      the number of seconds accounted for by this
seconds    function alone.  This is the major sort for this
           listing.

calls      the number of times this function was invoked, if
           this function is profiled, else blank.

 self      the average number of milliseconds spent in this
ms/call    function per call, if this function is profiled,
       else blank.

 total     the average number of milliseconds spent in this
ms/call    function and its descendents per call, if this
       function is profiled, else blank.

name       the name of the function.  This is the minor sort
           for this listing. The index shows the location of
       the function in the gprof listing. If the index is
       in parenthesis it shows where it would appear in
       the gprof listing if it were to be printed.
<0x0c>
Copyright (C) 2012-2020 Free Software Foundation, Inc.

Copying and distribution of this file, with or without modification,
are permitted in any medium without royalty provided the copyright
notice and this notice are preserved.
```

G = 90
Cubic Spline
Time taken = 0.48 Seconds

```
Flat profile:

Each sample counts as 0.01 seconds.
 no time accumulated

  %   cumulative   self              self     total
 time   seconds   seconds    calls  Ts/call  Ts/call  name

 %         the percentage of the total running time of the
time       program used by this function.

cumulative a running sum of the number of seconds accounted
 seconds   for by this function and those listed above it.

 self      the number of seconds accounted for by this
seconds    function alone.  This is the major sort for this
           listing.

calls      the number of times this function was invoked, if
           this function is profiled, else blank.

 self      the average number of milliseconds spent in this
ms/call    function per call, if this function is profiled,
       else blank.

 total     the average number of milliseconds spent in this
ms/call    function and its descendents per call, if this
       function is profiled, else blank.

name       the name of the function.  This is the minor sort
           for this listing. The index shows the location of
       the function in the gprof listing. If the index is
       in parenthesis it shows where it would appear in
       the gprof listing if it were to be printed.
<0x0c>
Copyright (C) 2012-2020 Free Software Foundation, Inc.

Copying and distribution of this file, with or without modification,
are permitted in any medium without royalty provided the copyright
notice and this notice are preserved.
```

G = 90
Linear spline
Unfortunately (or fortunately) the time it takes is less than the least count. So the output is shown as "No time accumulated"

```
Flat profile:

Each sample counts as 0.01 seconds.
  %   cumulative   self              self     total
 time   seconds   seconds    calls  Ts/call  Ts/call  name
100.17      0.04     0.04                             main

 %         the percentage of the total running time of the
time       program used by this function.

cumulative a running sum of the number of seconds accounted
 seconds   for by this function and those listed above it.

 self      the number of seconds accounted for by this
seconds    function alone.  This is the major sort for this
           listing.

calls      the number of times this function was invoked, if
           this function is profiled, else blank.

 self      the average number of milliseconds spent in this
ms/call    function per call, if this function is profiled,
       else blank.

 total     the average number of milliseconds spent in this
ms/call    function and its descendents per call, if this
       function is profiled, else blank.

name       the name of the function.  This is the minor sort
           for this listing. The index shows the location of
       the function in the gprof listing. If the index is
       in parenthesis it shows where it would appear in
       the gprof listing if it were to be printed.
<0x0c>
Copyright (C) 2012-2020 Free Software Foundation, Inc.

Copying and distribution of this file, with or without modification,
are permitted in any medium without royalty provided the copyright
notice and this notice are preserved.
```

G = 200
Cubic Spline
Time taken = 0.04 Seconds

```
57
58
59  Flat profile:
60
61  Each sample counts as 0.01 seconds.
62   no time accumulated
63
64    %   cumulative   self              self     total
65   time   seconds   seconds    calls  Ts/call  Ts/call  name
66
67   %         the percentage of the total running time of the
68  time       program used by this function.
69
70  cumulative a running sum of the number of seconds accounted
71   seconds   for by this function and those listed above it.
72
73   self      the number of seconds accounted for by this
74  seconds    function alone.  This is the major sort for this
75             listing.
76
77  calls      the number of times this function was invoked, if
78             this function is profiled, else blank.
79
80   self      the average number of milliseconds spent in this
81  ms/call    function per call, if this function is profiled,
82         else blank.
83
84   total     the average number of milliseconds spent in this
85  ms/call    function and its descendents per call, if this
86         function is profiled, else blank.
87
88  name       the name of the function.  This is the minor sort
89             for this listing. The index shows the location of
90         the function in the gprof listing. If the index is
91         in parenthesis it shows where it would appear in
92         the gprof listing if it were to be printed.
93  <0x0c>
94  Copyright (C) 2012-2020 Free Software Foundation, Inc.
95
96  Copying and distribution of this file, with or without modification,
97  are permitted in any medium without royalty provided the copyright
98  notice and this notice are preserved.
99
100
101
102
103
104
105
106
107
108
109
110
111
```

G = 200
Linear spline
Unfortunately (or fortunately) the time it takes is less than the least count. So the output is shown as "No time accumulated"

```
111
112
113
114 Flat profile:
115
116 Each sample counts as 0.01 seconds.
117  no time accumulated
118
119   %   cumulative   self              self     total
120  time   seconds   seconds    calls  Ts/call  Ts/call  name
121
122  %         the percentage of the total running time of the
123 time       program used by this function.
124
125 cumulative a running sum of the number of seconds accounted
126  seconds   for by this function and those listed above it.
127
128  self      the number of seconds accounted for by this
129 seconds    function alone.  This is the major sort for this
130           listing.
131
132 calls      the number of times this function was invoked, if
133           this function is profiled, else blank.
134
135  self      the average number of milliseconds spent in this
136 ms/call    function per call, if this function is profiled,
137        else blank.
138
139  total     the average number of milliseconds spent in this
140 ms/call    function and its descendents per call, if this
141        function is profiled, else blank.
142
143 name       the name of the function.  This is the minor sort
144           for this listing. The index shows the location of
145        the function in the gprof listing. If the index is
146        in parenthesis it shows where it would appear in
147        the gprof listing if it were to be printed.
148 <0x0c>
149 Copyright (C) 2012-2020 Free Software Foundation, Inc.
150
151 Copying and distribution of this file, with or without modification,
152 are permitted in any medium without royalty provided the copyright
153 notice and this notice are preserved.
154
```

G = 700
Cubic Spline
In this case, for a large
G like 700, the time taken
is again not calculated
as it is less than the least count

```
111
112
113
114  Flat profile:
115
116  Each sample counts as 0.01 seconds.
117   no time accumulated
118
119    %   cumulative   self              self     total
120   time   seconds   seconds    calls  Ts/call  Ts/call  name
121
122   %         the percentage of the total running time of the
123  time       program used by this function.
124
125  cumulative a running sum of the number of seconds accounted
126   seconds   for by this function and those listed above it.
127
128   self      the number of seconds accounted for by this
129  seconds    function alone.  This is the major sort for this
130             listing.
131
132  calls      the number of times this function was invoked, if
133             this function is profiled, else blank.
134
135   self      the average number of milliseconds spent in this
136  ms/call    function per call, if this function is profiled,
137         else blank.
138
139   total     the average number of milliseconds spent in this
140  ms/call    function and its descendents per call, if this
141         function is profiled, else blank.
142
143  name       the name of the function.  This is the minor sort
144             for this listing. The index shows the location of
145         the function in the gprof listing. If the index is
146         in parenthesis it shows where it would appear in
147         the gprof listing if it were to be printed.
148  <0x0c>
149  Copyright (C) 2012-2020 Free Software Foundation, Inc.
150
151  Copying and distribution of this file, with or without modification,
152  are permitted in any medium without royalty provided the copyright
153  notice and this notice are preserved.
154
```

G = 700
Linear spline
Unfortunately (or fortunately) the time it takes is less than the least count. So the output is shown as "No time accumulated"

From the graphs and the profiling data, we can say:

1. **Linear spline is faster than Cubic spline**
2. **But Cubic spline is more accurate than Linear and can also fit oscillating data reasonably well**