

# Parallel Algorithms: HW2 Screenshots

## Jonathan Land

The screenshot shows the Xcode interface with two files open:

- HW2\_posix.c** (Left Window):
 

```

7
8  /*
9   * Jonathan Land
10  * 2/6/18
11 */
12
13 int NUM_THREADS = 1;
14
15 struct args_struct
16 {
17     double* arr;
18     int position;
19     int size;
20     double max;
21     double sum;
22     double standardDeviation;
23     double mean;
24     double w;
25 };
26
27 void* randomArrayThread(void* args)
28 {
29     //casting args into arguments and casting it
30     //can have arguments before we fill up to send
31     struct args_struct* arguments = args;
32
33     //method to change up the values each function call
34     //srand((unsigned int)time(NULL));
35
36     //give one memory space
37     double* p = arguments->arr + arguments->position;
38
39     for(int i = arguments->position; i < arguments->position + arguments->size;
40         i++)
41     {
42         *p = ((double) rand() / RAND_MAX) * arguments->max;
43         p++;
44     }
45     return NULL;
46 }
```
- HW2\_posix\_main.c** (Right Window):
 

```

//test arraySize for class = 100000000
//may need to change in random_array also
int arraySize = 100000000;

float smoothA = .10;
double arr[arraySize];

printf("      Homework 2 Synopsis: \n");
printf("===== \n");
printf("Total number of threads: %d\n", NUM_THREADS);
printf("Size of array: %d\n", arraySize);
printf("Weight of smooth: %f\n", smoothA);
printf("\n");
printf("===== \n");

//timing for random array
printf("\n");
printf("Creating the Array \n");
printf("\n");

struct timeval tv1, tv2;
gettimeofday(&tv1, NULL);
random_array(arr, arraySize, 100000000);
gettimeofday(&tv2, NULL);

printf("\n");
printf(">>Total time to create array: %f seconds\n", (double) (tv2.tv_usec - tv1.
    + (double) (tv2.tv_sec - tv1.tv_sec)));
printf("\n");
printf("===== \n");

//use to check range of values
/*
for (int i = 0; i < arraySize; i++)
{
    printf("%f ", arr[i]);
}
*/
```

The screenshot shows a terminal window with the following output:

```

sland — jland@qbert:~ — ssh jland@qbert.utc.edu — 123x32
Last login: Mon Feb  5 21:43:33 on ttys000
[jlandMP:-] jland$ ssh jland qbert@utc.edu
ssh: Could not resolve hostname jland: nodename nor servname provided, or not known
[jlandMP:-] jland$ ssh jland@qbert.utc.edu
[jland@qbert.utc.edu's password:
Last login: Thu Feb  1 14:37:45 2018 from jlandmp.wireless.utc.edu
#####
##### Welcome to qbert; if you use this cluster for published research, please use
##### the following excerpt to cite this cluster:
#####
##### This material is based upon work supported by the National Science Foundation
##### under Major Research Instrumentation (MRI) Grant No. 1229213. Any opinions,
##### findings, and conclusions or recommendations expressed in this material are
##### those of the authors and do not necessarily reflect the views of the National
##### Science Foundation
#####
[jland@qbert ~]$
```

The terminal window also displays the source code of `HW2_posix_main.c` at the bottom.

```
Terminal Shell Edit View Window Help
sland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50
HW2_posix_main.c:36:38: note: each undeclared identifier is reported only once for each function it appears in
[jland@qbert PosixProgram]$ gcc -g -o PosixProgramHW2 HW2_posix.c HW2_posix_main.c -lpthread -lm
[jland@qbert PosixProgram]$ ./PosixProgramHW2
Homework 2 Synopsis:
=====
[Total number of threads: 1
[Size of array: 100000000
Weight of smooth: 0.100000
=====
Creating the Array
Arguments: 100000000 10.000000 0
Attaching to thread 0
>>Total time to create array: 2.237902 seconds
=====
Summing the Array
Arguments: 100000000 0
Attaching to thread 0
>>Sum of the array: 500010842.857140
>>Total time to calculate the sum of the array: 0.303263 seconds
=====
Standard Deviation of the Array(parallelized)
Arguments: 100000000 0.000000 0 5.000108
Attaching to thread 0
>>Standard deviation of parallelized array: 2.886603
>>Total time to calculate parallelized standard deviation: 2.032824 seconds
>>Standard deviation of original/serial array: 2.886603
>>Result of original/serial standard deviation (i.e., HW1): 2.886603
>>Total time to calculate original/serial standard deviation: 2.069867 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 1.115867 seconds
[jland@qbert PosixProgram]$
```

```
Terminal Shell Edit View Window Help
sland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50
Homework 2 Synopsis:
To =====
-- Total number of threads: 2
-- Size of array: 100000000
## Weight of smooth: 0.100000
##
##
## =====
## Creating the Array
## Arguments: 50000000 10.000000 0
## Arguments: 50000000 10.000000 50000000
## Attaching to thread 0
## Attaching to thread 1
[d] >>Total time to create array: 30.901530 seconds
##
##
[d] Summing the Array
[d]
## Arguments: 50000000 0
[d] Arguments: 50000000 50000000
[d] Attaching to thread 0
[d] Attaching to thread 1
Po
[d] >>Sum of the array: 500010842.857265
/d >>Total time to calculate the sum of the array: 0.152750 seconds
de
/d =====
ma
ma Standard Deviation of the Array(parallelized)
[d]
[d] Arguments: 50000000 0.000000 0 5.000108
[d] Arguments: 50000000 0.000000 50000000 5.000108
[d] Attaching to thread 0
[d] Attaching to thread 1
>>Standard deviation of parallelized array: 2.886603
>>Total time to calculate parallelized standard deviation: 1.174124 seconds
>>Standard deviation of original/serial array: 2.886603
>>Result of original/serial standard deviation (i.e., HW1): 2.886603
>>Total time to calculate original/serial standard deviation: 2.046666 seconds
=====
Smoothing the Array
```

```
Terminal Shell Edit View Window Help
sland — jlaland@qbert:~/PosixProgram — ssh jlaland@qbert.utc.edu — 160x50
Total number of threads: 2
## Size of array: 100000000
## Weight of smooth: 0.100000
##
## =====
## Creating the Array
##
## Arguments: 50000000 10.000000
## Arguments: 50000000 10.000000 50000000
## Attaching to thread 0
## Attaching to thread 1
##
## >>>Total time to create array: 30.901530 seconds
[J1] =====
## Summing the Array
[J1] Arguments: 50000000 0
[J1] Arguments: 50000000 50000000
[J1] Attaching to thread 0
## Attaching to thread 1
[J1]
[J1] >>>Sum of the array: 500010842.857265
[J1] >>>Total time to calculate the sum of the array: 0.152750 seconds
Po
[J1] =====
/d
de Standard Deviation of the Array(parallelized)
/d
ma Arguments: 50000000 0.000000 0 5.000108
ma Arguments: 50000000 0.000000 50000000 5.000108
jl Attaching to thread 0
[J1] Attaching to thread 1
[J1]
[J1] >>>Standard deviation of parallelized array: 2.886603
[J1] >>>Total time to calculate parallelized standard deviation: 1.174124 seconds
##
>>>Standard deviation of original/serial array: 2.886603
>>>Result of original/serial standard deviation (i.e., HV1): 2.886603
>>>Total time to calculate original/serial standard deviation: 2.046666 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 0.553350 seconds
[jlaland@qbert PosixProgram]$
```

```
Terminal Shell Edit View Window Help
sland — jlaland@qbert:~/PosixProgram — ssh jlaland@qbert.utc.edu — 160x50
[[jlaland@qbert PosixProgram]$ ./PosixProgramHW2
Yo
Homework 2 Synopsis:
=====
## Total number of threads: 4
## Size of array: 100000000
## Weight of smooth: 0.100000
##
## =====
## Creating the Array
##
## Arguments: 25000000 10.000000 0
## Arguments: 25000000 10.000000 25000000
[J1] Arguments: 25000000 10.000000 50000000
[J1] Arguments: 25000000 10.000000 75000000
## Attaching to thread 0
## Attaching to thread 1
## Attaching to thread 2
[J1] Attaching to thread 3
[J1]
[J1] >>>Total time to create array: 18.154045 seconds
mo
[J1] =====
[J1]
[J1] Summing the Array
Po
[J1] Arguments: 25000000 0
/d Arguments: 25000000 25000000
de Arguments: 25000000 50000000
/d Arguments: 25000000 75000000
ma Attaching to thread 0
ma Attaching to thread 1
jl Attaching to thread 2
[J1] Attaching to thread 3
[J1]
[J1] >>>Sum of the array: 500010842.857395
[J1] >>>Total time to calculate the sum of the array: 0.161487 seconds
=====
Standard Deviation of the Array(parallelized)
Arguments: 25000000 0.000000 0 5.000108
Arguments: 25000000 0.000000 25000000 5.000108
Arguments: 25000000 0.000000 50000000 5.000108
Arguments: 25000000 0.000000 75000000 5.000108
Attaching to thread 0
Attaching to thread 1
Attaching to thread 2
```

```
Terminal Shell Edit View Window Help sland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50
Attaching to thread 0
Yo Attaching to thread 1
Attaching to thread 2
== Attaching to thread 3
==
## >>Total time to create array: 10.154045 seconds
==
=====
== Summing the Array
==
## Arguments: 25000000 0
## Arguments: 25000000 25000000
## Arguments: 25000000 50000000
## Arguments: 25000000 75000000
[J] Attaching to thread 0
== Attaching to thread 1
## Attaching to thread 2
## Attaching to thread 3
[J]
[J] >>Sum of the array: 500010842.657395
[J] >>Total time to calcuate the sum of the array: 0.161487 seconds
no
[J] =====
[J]
[J] Standard Deviation of the Array(parallelized)
Po
[J] Arguments: 25000000 0.000000 0 5.000108
/d Arguments: 25000000 0.000000 25000000 5.000108
de Arguments: 25000000 0.000000 50000000 5.000108
/d Arguments: 25000000 0.000000 75000000 5.000108
ma Attaching to thread 0
ma Attaching to thread 1
jl Attaching to thread 2
[J] Attaching to thread 3
[J]
[J] >>Standard deviation of parallelized array: 2.886603
[J] >>Total time to calcuate parallelized standard deviation: 1.297871 seconds
>>Standard deviation of original/serial array: 2.886603
>>Result of original/serial standard deviation (i.e., HV1): 2.886603
>>Total time to calculate original/serial standard deviation: 2.441909 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 0.590814 seconds
[jland@qbert PosixProgram]$
```

```
Terminal Shell Edit View Window Help sland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50
Homework 2 Synopsis:
Yo =====
== Total number of threads: 8
## Size of array: 100000000
## Weight of smooth: 0.100000
==
==
== Creating the Array
##
## Arguments: 12500000 10.000000 0
## Arguments: 12500000 10.000000 12500000
## Arguments: 12500000 10.000000 25000000
## Arguments: 12500000 10.000000 37500000
[J] Arguments: 12500000 10.000000 50000000
## Arguments: 12500000 10.000000 62500000
## Arguments: 12500000 10.000000 75000000
## Arguments: 12500000 10.000000 87500000
[D] Attaching to thread 0
[J] Attaching to thread 1
[J] Attaching to thread 2
no Attaching to thread 3
[J] Attaching to thread 4
[J] Attaching to thread 5
[J] Attaching to thread 6
Po Attaching to thread 7
[J]
/d >>Total time to create array: 23.784732 seconds
de
/d =====
ma
ma Summing the Array
jl
[J] Arguments: 12500000 0
[J] Arguments: 12500000 12500000
[J] Arguments: 12500000 25000000
[J] Arguments: 12500000 37500000
Arguments: 12500000 50000000
Arguments: 12500000 62500000
Arguments: 12500000 75000000
Arguments: 12500000 87500000
Attaching to thread 0
Attaching to thread 1
Attaching to thread 2
Attaching to thread 3
Attaching to thread 4
Attaching to thread 5
Attaching to thread 6
Attaching to thread 7
```

```
Terminal Shell Edit View Window Help
sland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50
Attaching to thread 4
Yo Attaching to thread 5
Attaching to thread 6
== Attaching to thread 7
==
## >>Total time to create array: 23.784732 seconds
==
== =====
== Summing the Array
==
## Arguments: 12500000 0
## Arguments: 12500000 12500000
## Arguments: 12500000 25000000
## Arguments: 12500000 37500000
[J1] Arguments: 12500000 50000000
## Arguments: 12500000 62500000
## Arguments: 12500000 75000000
## Arguments: 12500000 87500000
[D] Attaching to thread 8
[J1] Attaching to thread 1
[J1] Attaching to thread 2
mo Attaching to thread 3
[J1] Attaching to thread 4
[J1] Attaching to thread 5
[J1] Attaching to thread 6
Po Attaching to thread 7
[J1]
/d >>Sum of the array: 500010842.857337
de >>Total time to calculate the sum of the array: 0.112772 seconds
/d
ma =====
ma
jl Standard Deviation of the Array(parallelized)
[J1]
[J1] Arguments: 12500000 0.000000 0 5.000108
[J1] Arguments: 12500000 0.000000 12500000 5.000108
[J1] Arguments: 12500000 0.000000 25000000 5.000108
[J1] Arguments: 12500000 0.000000 37500000 5.000108
Arguments: 12500000 0.000000 50000000 5.000108
Arguments: 12500000 0.000000 62500000 5.000108
Arguments: 12500000 0.000000 75000000 5.000108
Arguments: 12500000 0.000000 87500000 5.000108
Attaching to thread 8
Attaching to thread 1
Attaching to thread 2
Attaching to thread 3
Attaching to thread 4
Attaching to thread 5
Attaching to thread 6
[J1] Attaching to thread 7
==
== >>Sum of the array: 500010842.857337
== >>Total time to calculate the sum of the array: 0.112772 seconds
[D]
[J1] =====
## Standard Deviation of the Array(parallelized)
[D] Arguments: 12500000 62500000
## Arguments: 12500000 75000000
## Arguments: 12500000 87500000
== Attaching to thread 8
== Attaching to thread 1
## Attaching to thread 2
## Attaching to thread 3
## Attaching to thread 4
## Attaching to thread 5
## Attaching to thread 6
## Attaching to thread 7
##
## >>Sum of the array: 500010842.857337
## >>Total time to calculate the sum of the array: 0.112772 seconds
[J1]
[J1] =====
## Standard Deviation of the Array(parallelized)
[D] Arguments: 12500000 0.000000 0 5.000108
[J1] Arguments: 12500000 0.000000 12500000 5.000108
[J1] Arguments: 12500000 0.000000 25000000 5.000108
[J1] Arguments: 12500000 0.000000 37500000 5.000108
[J1] Arguments: 12500000 0.000000 50000000 5.000108
[J1] Arguments: 12500000 0.000000 62500000 5.000108
[J1] Arguments: 12500000 0.000000 75000000 5.000108
Po Arguments: 12500000 0.000000 87500000 5.000108
[J1] Attaching to thread 8
/d Attaching to thread 1
de Attaching to thread 2
/d Attaching to thread 3
ma Attaching to thread 4
ma Attaching to thread 5
jl Attaching to thread 6
[J1] Attaching to thread 7
[J1]
[J1] >>Standard deviation of parallelized array: 2.886603
[J1] >>Total time to calculate parallelized standard deviation: 1.037647 seconds
>>Standard deviation of original/serial array: 2.886603
>>Result of original/serial standard deviation (i.e., HV1): 2.886603
>>Total time to calculate original/serial standard deviation: 2.038648 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 0.421243 seconds
[J1]@qbert PosixProgram$
```

```
Terminal Shell Edit View Window Help
sland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50
Attaching to thread 4
Yo Attaching to thread 5
Attaching to thread 6
== Attaching to thread 7
==
## >>Total time to create array: 23.784732 seconds
==
== =====
== Summing the Array
==
## Arguments: 12500000 0
## Arguments: 12500000 12500000
## Arguments: 12500000 25000000
## Arguments: 12500000 37500000
[J1] Arguments: 12500000 50000000
## Arguments: 12500000 62500000
## Arguments: 12500000 75000000
## Arguments: 12500000 87500000
[D] Attaching to thread 8
[J1] Attaching to thread 1
[J1] Attaching to thread 2
mo Attaching to thread 3
[J1] Attaching to thread 4
[J1] Attaching to thread 5
[J1] Attaching to thread 6
Po Attaching to thread 7
[J1]
/d >>Sum of the array: 500010842.857337
de >>Total time to calculate the sum of the array: 0.112772 seconds
/d
ma =====
ma
jl Standard Deviation of the Array(parallelized)
[J1]
[J1] Arguments: 12500000 0.000000 0 5.000108
[J1] Arguments: 12500000 0.000000 12500000 5.000108
[J1] Arguments: 12500000 0.000000 25000000 5.000108
[J1] Arguments: 12500000 0.000000 37500000 5.000108
[J1] Arguments: 12500000 0.000000 50000000 5.000108
[J1] Arguments: 12500000 0.000000 62500000 5.000108
[J1] Arguments: 12500000 0.000000 75000000 5.000108
Po Arguments: 12500000 0.000000 87500000 5.000108
[J1] Attaching to thread 8
/d Attaching to thread 1
de Attaching to thread 2
/d Attaching to thread 3
ma Attaching to thread 4
ma Attaching to thread 5
jl Attaching to thread 6
[J1] Attaching to thread 7
[J1]
[J1] >>Standard deviation of parallelized array: 2.886603
[J1] >>Total time to calculate parallelized standard deviation: 1.037647 seconds
>>Standard deviation of original/serial array: 2.886603
>>Result of original/serial standard deviation (i.e., HV1): 2.886603
>>Total time to calculate original/serial standard deviation: 2.038648 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 0.421243 seconds
[J1]@qbert PosixProgram$
```

```
Terminal Shell Edit View Window Help sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Homework 2 Synopsis:
To =====
-- Total number of threads: 8
-- Size of array: 10000000
## Weight of smooth: 0.100000
##
-- =====
-- Creating the Array
--
[J1 Arguments: 12500000 10.000000 0
[J1 Arguments: 12500000 10.000000 12500000
[J1 Arguments: 12500000 10.000000 25000000
[J1 Arguments: 12500000 10.000000 37500000
[J1 Arguments: 12500000 10.000000 50000000
[J1 Arguments: 12500000 10.000000 62500000
## Arguments: 12500000 10.000000 75000000
Arguments: 12500000 10.000000 87500000
[D] Attaching to thread 0
[J1 Attaching to thread 1
[J1 Attaching to thread 2
[m] Attaching to thread 3
[J1 Attaching to thread 4
[J1 Attaching to thread 5
[J1 Attaching to thread 6
[Po] Attaching to thread 7
[D]
/d >>Total time to create array: 23.784732 seconds
de
/d =====
[ma]
[ma] Summing the Array
[jl]
[J1 Arguments: 12500000 0
[J1 Arguments: 12500000 12500000
[J1 Arguments: 12500000 25000000
[J1 Arguments: 12500000 37500000
[J1 Arguments: 12500000 50000000
Arguments: 12500000 62500000
Arguments: 12500000 75000000
Arguments: 12500000 87500000
Attaching to thread 0
Attaching to thread 1
Attaching to thread 2
Attaching to thread 3
Attaching to thread 4
Attaching to thread 5
Attaching to thread 6
Attaching to thread 7
[ma]
```

```
Terminal Shell Edit View Window Help sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Homework 2 Synopsis:
To =====
-- Total number of threads: 16
-- Size of array: 10000000
## Weight of smooth: 0.100000
##
-- =====
-- Creating the Array
--
[J1 Arguments: 6250000 10.000000 0
[J1 Arguments: 6250000 10.000000 6250000
[J1 Arguments: 6250000 10.000000 12500000
[J1 Arguments: 6250000 10.000000 18750000
[J1 Arguments: 6250000 10.000000 25000000
[J1 Arguments: 6250000 10.000000 31250000
## Arguments: 6250000 10.000000 37500000
Arguments: 6250000 10.000000 43750000
[D] Arguments: 6250000 10.000000 50000000
[J1 Arguments: 6250000 10.000000 56250000
[J1 Arguments: 6250000 10.000000 62500000
[m] Arguments: 6250000 10.000000 68750000
[J1 Arguments: 6250000 10.000000 75000000
[J1 Arguments: 6250000 10.000000 81250000
[J1 Arguments: 6250000 10.000000 87500000
Po Arguments: 6250000 10.000000 93750000
[D] Attaching to thread 0
[d] Attaching to thread 1
[de] Attaching to thread 2
[d] Attaching to thread 3
[ma] Attaching to thread 4
[ma] Attaching to thread 5
[jl] Attaching to thread 6
[jl] Attaching to thread 7
[jl] Attaching to thread 8
[jl] Attaching to thread 9
[jl] Attaching to thread 10
Attaching to thread 11
Attaching to thread 12
Attaching to thread 13
Attaching to thread 14
Attaching to thread 15
[ma]
>>Total time to create array: 28.963296 seconds
=====
Summing the Array
```

```
Terminal Shell Edit View Window Help sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Attaching to thread 4
Yo Attaching to thread 5
Attaching to thread 6
== Attaching to thread 7
== Attaching to thread 8
## Attaching to thread 9
## Attaching to thread 10
## Attaching to thread 11
## Attaching to thread 12
## Attaching to thread 13
## Attaching to thread 14
## Attaching to thread 15
##
>>>Total time to create array: 28.963296 seconds
[J1] ****
## Summing the Array
[D] Arguments: 62500000 8
[J1] Arguments: 62500000 62500000
[J1] Arguments: 62500000 12500000
[m] Arguments: 62500000 18750000
[J1] Arguments: 62500000 25000000
[J1] Arguments: 62500000 31250000
[J1] Arguments: 62500000 37500000
[Po] Arguments: 62500000 43750000
[J1] Arguments: 62500000 50000000
[d] Arguments: 62500000 56250000
[de] Arguments: 62500000 62500000
[d/] Arguments: 62500000 68750000
[ma] Arguments: 62500000 75000000
[ma] Arguments: 62500000 81250000
[jl] Arguments: 62500000 87500000
[J1] Arguments: 62500000 93750000
[J1] Attaching to thread 8
[J1] Attaching to thread 1
[J1] Attaching to thread 2
[J1] Attaching to thread 3
[J1] Attaching to thread 4
[J1] Attaching to thread 5
[J1] Attaching to thread 6
[J1] Attaching to thread 7
[J1] Attaching to thread 8
[J1] Attaching to thread 9
[J1] Attaching to thread 10
[J1] Attaching to thread 11
[J1] Attaching to thread 12
[J1] Attaching to thread 13
[J1] Attaching to thread 14
[J1] Attaching to thread 15
[J1]
[J1] ****
[J1] >>Sum of the array: 500018842.857343
[m] >>Total time to calculate the sum of the array: 0.117736 seconds
[J1]
[J1] ****
[J1]
Po Standard Deviation of the Array(parallelized)
[J1]
[d] Arguments: 62500000 0.000000 0 5.000108
[de] Arguments: 62500000 0.000000 62500000 5.000108
[d/] Arguments: 62500000 0.000000 12500000 5.000108
[ma] Arguments: 62500000 0.000000 18750000 5.000108
[ma] Arguments: 62500000 0.000000 25000000 5.000108
[J1] Arguments: 62500000 0.000000 31250000 5.000108
[J1] Arguments: 62500000 0.000000 37500000 5.000108
[J1] Arguments: 62500000 0.000000 43750000 5.000108
[J1] Arguments: 62500000 0.000000 50000000 5.000108
[J1] Arguments: 62500000 0.000000 56250000 5.000108
[J1] Arguments: 62500000 0.000000 62500000 5.000108
Arguments: 62500000 0.000000 68750000 5.000108
Arguments: 62500000 0.000000 75000000 5.000108
Arguments: 62500000 0.000000 81250000 5.000108
Arguments: 62500000 0.000000 87500000 5.000108
Arguments: 62500000 0.000000 93750000 5.000108
Attaching to thread 8
Attaching to thread 1
Attaching to thread 2
Attaching to thread 3
Attaching to thread 4
Attaching to thread 5
```

```
Terminal Shell Edit View Window Help sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Arguments: 62500000 75000000
Yo Arguments: 62500000 81250000
Arguments: 62500000 87500000
== Arguments: 62500000 93750000
== Attaching to thread 8
## Attaching to thread 1
## Attaching to thread 2
## Attaching to thread 3
## Attaching to thread 4
## Attaching to thread 5
## Attaching to thread 6
## Attaching to thread 7
## Attaching to thread 8
## Attaching to thread 9
## Attaching to thread 10
## Attaching to thread 11
## Attaching to thread 12
## Attaching to thread 13
## Attaching to thread 14
## Attaching to thread 15
[J1]
[J1] ****
[J1] >>Sum of the array: 500018842.857343
[m] >>Total time to calculate the sum of the array: 0.117736 seconds
[J1]
[J1] ****
[J1]
Po Standard Deviation of the Array(parallelized)
[J1]
[d] Arguments: 62500000 0.000000 0 5.000108
[de] Arguments: 62500000 0.000000 62500000 5.000108
[d/] Arguments: 62500000 0.000000 12500000 5.000108
[ma] Arguments: 62500000 0.000000 18750000 5.000108
[ma] Arguments: 62500000 0.000000 25000000 5.000108
[J1] Arguments: 62500000 0.000000 31250000 5.000108
[J1] Arguments: 62500000 0.000000 37500000 5.000108
[J1] Arguments: 62500000 0.000000 43750000 5.000108
[J1] Arguments: 62500000 0.000000 50000000 5.000108
[J1] Arguments: 62500000 0.000000 56250000 5.000108
[J1] Arguments: 62500000 0.000000 62500000 5.000108
Arguments: 62500000 0.000000 68750000 5.000108
Arguments: 62500000 0.000000 75000000 5.000108
Arguments: 62500000 0.000000 81250000 5.000108
Arguments: 62500000 0.000000 87500000 5.000108
Arguments: 62500000 0.000000 93750000 5.000108
Attaching to thread 8
Attaching to thread 1
Attaching to thread 2
Attaching to thread 3
Attaching to thread 4
Attaching to thread 5
```

Terminal Shell Edit View Window Help

jland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50

```

Yo Standard Deviation of the Array(parallelized)
-- Arguments: 6250000 8.000000 0 5.000108
-- Arguments: 6250000 8.000000 6250000 5.000108
## Arguments: 6250000 8.000000 12500000 5.000108
## Arguments: 6250000 8.000000 18750000 5.000108
## Arguments: 6250000 8.000000 25000000 5.000108
## Arguments: 6250000 8.000000 31250000 5.000108
## Arguments: 6250000 8.000000 37500000 5.000108
## Arguments: 6250000 8.000000 43750000 5.000108
## Arguments: 6250000 8.000000 50000000 5.000108
## Arguments: 6250000 8.000000 62500000 5.000108
## Arguments: 6250000 8.000000 75000000 5.000108
## Arguments: 6250000 8.000000 87500000 5.000108
## Arguments: 6250000 8.000000 95750000 5.000108
[J] Attaching to thread 0
[J] Attaching to thread 1
[J] Attaching to thread 2
[J] Attaching to thread 3
[J] Attaching to thread 4
[J] Attaching to thread 5
[J] Attaching to thread 6
[P] Attaching to thread 7
[J] Attaching to thread 8
[d] Attaching to thread 9
[de] Attaching to thread 10
[d] Attaching to thread 11
[ma] Attaching to thread 12
[ma] Attaching to thread 13
[jl] Attaching to thread 14
[J] Attaching to thread 15
[J]
[J] >>Standard deviation of parallelized array: 2.886603
[J] >>Total time to calcuate parallelized standard deviation: 1.735591 seconds
    >>Standard deviation of original/serial array: 2.886603
    >>Result of original/serial standard deviation (i.e., HV1): 2.886603
    >>Total time to calculate original/serial standard deviation: 5.389868 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 0.204588 seconds
[jland@qbert PosixProgram]$
```

Terminal Shell Edit View Window Help

jland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50

```

Homework 2 Synopsis:
Yo =====
-- Total number of threads: 32
-- Size of array: 100000000
## Weight of smooth: 0.100000
##
-- Creating the Array
##
-- Arguments: 3125000 18.000000 0
-- Arguments: 3125000 18.000000 3125000
## Arguments: 3125000 18.000000 6250000
[J] Arguments: 3125000 18.000000 9375000
[J] Arguments: 3125000 18.000000 12500000
## Arguments: 3125000 18.000000 15625000
## Arguments: 3125000 18.000000 18750000
## Arguments: 3125000 18.000000 21875000
[J] Arguments: 3125000 18.000000 25000000
[J] Arguments: 3125000 18.000000 28125000
[J] Arguments: 3125000 18.000000 31250000
[mo] Arguments: 3125000 18.000000 34375000
[J] Arguments: 3125000 18.000000 37500000
[J] Arguments: 3125000 18.000000 40625000
[J] Arguments: 3125000 18.000000 43750000
[Po] Arguments: 3125000 18.000000 46875000
[J] Arguments: 3125000 18.000000 50000000
[d] Arguments: 3125000 18.000000 53125000
[de] Arguments: 3125000 18.000000 56250000
[d/] Arguments: 3125000 18.000000 59375000
[ma] Arguments: 3125000 18.000000 62500000
[ma] Arguments: 3125000 18.000000 65625000
[jl] Arguments: 3125000 18.000000 68750000
[J] Arguments: 3125000 18.000000 71875000
[J] Arguments: 3125000 18.000000 75000000
[J] Arguments: 3125000 18.000000 78125000
[J] Arguments: 3125000 18.000000 81250000
Arguments: 3125000 18.000000 84375000
Arguments: 3125000 18.000000 87500000
Arguments: 3125000 18.000000 90625000
Arguments: 3125000 18.000000 93750000
Arguments: 3125000 18.000000 96875000
Attaching to thread 0
Attaching to thread 1
Attaching to thread 2
Attaching to thread 3
Attaching to thread 4
Attaching to thread 5
Attaching to thread 6
```

```
Terminal Shell Edit View Window Help
sland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50
Arguments: 31250000 18.000000 40625000
Arguments: 31250000 18.000000 43750000
Arguments: 31250000 18.000000 46875000
== Arguments: 31250000 18.000000 50000000
## Arguments: 31250000 18.000000 53125000
## Arguments: 31250000 18.000000 56250000
## Arguments: 31250000 18.000000 59375000
## Arguments: 31250000 18.000000 62500000
## Arguments: 31250000 18.000000 65625000
## Arguments: 31250000 18.000000 68750000
## Arguments: 31250000 18.000000 71875000
## Arguments: 31250000 18.000000 75000000
## Arguments: 31250000 18.000000 78125000
## Arguments: 31250000 18.000000 81250000
[!] Arguments: 31250000 18.000000 84375000
[!] Arguments: 31250000 18.000000 87500000
## Arguments: 31250000 18.000000 90625000
## Arguments: 31250000 18.000000 93750000
## Arguments: 31250000 18.000000 96875000
[!] Attaching to thread 0
[!] Attaching to thread 1
[!] Attaching to thread 2
[!] Attaching to thread 3
[!] Attaching to thread 4
[!] Attaching to thread 5
[!] Attaching to thread 6
[!] Attaching to thread 7
[!] Attaching to thread 8
/d Attaching to thread 9
de Attaching to thread 10
/d Attaching to thread 11
ma Attaching to thread 12
ma Attaching to thread 13
jl Attaching to thread 14
[jl] Attaching to thread 15
[jl] Attaching to thread 16
[jl] Attaching to thread 17
[jl] Attaching to thread 18
Attaching to thread 19
Attaching to thread 20
Attaching to thread 21
Attaching to thread 22
Attaching to thread 23
Attaching to thread 24
Attaching to thread 25
Attaching to thread 26
Attaching to thread 27
Attaching to thread 28
Attaching to thread 29
Attaching to thread 30
```

```
Attaching to thread 23
Attaching to thread 24
Attaching to thread 25
== Attaching to thread 26
== Attaching to thread 27
## Attaching to thread 28
== Attaching to thread 29
== Attaching to thread 30
== Attaching to thread 31
==

==>>Total time to create array: 14.052557 seconds
==

=====
Summing the Array
[jl]
== Arguments: 31250000 8
## Arguments: 31250000 31250000
Arguments: 31250000 62500000
Arguments: 31250000 93750000
[jl] Arguments: 31250000 125000000
[jl] Arguments: 31250000 156250000
[jl] Arguments: 31250000 187500000
[jl] Arguments: 31250000 218750000
[jl] Arguments: 31250000 250000000
[jl] Arguments: 31250000 281250000
P Arguments: 31250000 31250000
[jl] Arguments: 31250000 343750000
[jl] Arguments: 31250000 375000000
de Arguments: 31250000 406250000
/d Arguments: 31250000 437500000
ma Arguments: 31250000 468750000
ma Arguments: 31250000 500000000
jl Arguments: 31250000 531250000
[jl] Arguments: 31250000 562500000
[jl] Arguments: 31250000 593750000
[jl] Arguments: 31250000 625000000
[jl] Arguments: 31250000 656250000
Arguments: 31250000 687500000
Arguments: 31250000 718750000
Arguments: 31250000 750000000
Arguments: 31250000 781250000
Arguments: 31250000 812500000
Arguments: 31250000 843750000
Arguments: 31250000 875000000
Arguments: 31250000 906250000
Arguments: 31250000 937500000
Arguments: 31250000 968750000
Attaching to thread 8
Attaching to thread 1
```

```
Terminal Shell Edit View Window Help
sland — jlard@qbert:~/PosixProgram — ssh jlard@qbert.utc.edu — 160x50
=====
To Summing the Array
## Arguments: 3125000 8
## Arguments: 3125000 3125000
## Arguments: 3125000 6250000
## Arguments: 3125000 9375000
## Arguments: 3125000 12500000
## Arguments: 3125000 15625000
## Arguments: 3125000 18750000
## Arguments: 3125000 21875000
## Arguments: 3125000 25000000
## Arguments: 3125000 28125000
## Arguments: 3125000 31250000
[J] Arguments: 3125000 34375000
## Arguments: 3125000 37500000
## Arguments: 3125000 40625000
## Arguments: 3125000 43750000
## Arguments: 3125000 46875000
[J] Arguments: 3125000 50000000
[J] Arguments: 3125000 53125000
## Arguments: 3125000 56250000
[J] Arguments: 3125000 59375000
[J] Arguments: 3125000 62500000
[J] Arguments: 3125000 65625000
## Arguments: 3125000 68750000
[J] Arguments: 3125000 71875000
/d Arguments: 3125000 75000000
de Arguments: 3125000 78125000
/d Arguments: 3125000 81250000
## Arguments: 3125000 84375000
## Arguments: 3125000 87500000
[J] Arguments: 3125000 90625000
[J] Arguments: 3125000 93750000
[J] Arguments: 3125000 96875000
[J] Attaching to thread 0
[J] Attaching to thread 1
Attaching to thread 2
Attaching to thread 3
Attaching to thread 4
Attaching to thread 5
Attaching to thread 6
Attaching to thread 7
Attaching to thread 8
Attaching to thread 9
Attaching to thread 10
Attaching to thread 11
Attaching to thread 12
Attaching to thread 13
[J] Attaching to thread 14
Attaching to thread 15
## Attaching to thread 16
## Attaching to thread 17
## Attaching to thread 18
## Attaching to thread 19
## Attaching to thread 20
## Attaching to thread 21
## Attaching to thread 22
## Attaching to thread 23
## Attaching to thread 24
## Attaching to thread 25
## Attaching to thread 26
## Attaching to thread 27
[J] Attaching to thread 28
## Attaching to thread 29
## Attaching to thread 30
## Attaching to thread 31
[J] >>Sum of the array: 500010842.657334
[J] >>Total time to calculate the sum of the array: 8.026845 seconds
## =====
[J]
[J] Standard Deviation of the Array(parallelized)
Po
[J] Arguments: 3125000 0.000000 0 5.000108
/d Arguments: 3125000 0.000000 31250000 5.000108
de Arguments: 3125000 0.000000 62500000 5.000108
/d Arguments: 3125000 0.000000 93750000 5.000108
## Arguments: 3125000 -nan 12500000 5.000108
## Arguments: 3125000 0.000000 15625000 5.000108
[J] Arguments: 3125000 0.000000 18750000 5.000108
[J] Arguments: 3125000 0.000000 21875000 5.000108
[J] Arguments: 3125000 0.000000 25000000 5.000108
[J] Arguments: 3125000 0.000000 28125000 5.000108
[J] Arguments: 3125000 0.000000 31250000 5.000108
Arguments: 3125000 0.000000 34375000 5.000108
Arguments: 3125000 0.000000 37500000 5.000108
Arguments: 3125000 0.000000 40625000 5.000108
Arguments: 3125000 0.000000 43750000 5.000108
Arguments: 3125000 0.000000 46875000 5.000108
Arguments: 3125000 0.000000 50000000 5.000108
Arguments: 3125000 0.000000 53125000 5.000108
Arguments: 3125000 0.000000 56250000 5.000108
Arguments: 3125000 0.000000 59375000 5.000108
Arguments: 3125000 0.000000 62500000 5.000108
Arguments: 3125000 0.000000 65625000 5.000108
Arguments: 3125000 0.000000 68750000 5.000108
```

```
Terminal Shell Edit View Window Help
sland — jlard@qbert:~/PosixProgram — ssh jlard@qbert.utc.edu — 160x50
=====
To Summing the Array
## Arguments: 3125000 8
## Arguments: 3125000 3125000
## Arguments: 3125000 6250000
## Arguments: 3125000 9375000
## Arguments: 3125000 12500000
## Arguments: 3125000 15625000
## Arguments: 3125000 18750000
## Arguments: 3125000 21875000
## Arguments: 3125000 25000000
## Arguments: 3125000 28125000
## Arguments: 3125000 31250000
[J] Arguments: 3125000 34375000
## Arguments: 3125000 37500000
## Arguments: 3125000 40625000
## Arguments: 3125000 43750000
## Arguments: 3125000 46875000
[J] Arguments: 3125000 50000000
[J] Arguments: 3125000 53125000
## Arguments: 3125000 56250000
[J] Arguments: 3125000 59375000
[J] Arguments: 3125000 62500000
[J] Arguments: 3125000 65625000
[J] Arguments: 3125000 68750000
[J] Attaching to thread 0
[J] Attaching to thread 1
Attaching to thread 2
Attaching to thread 3
Attaching to thread 4
Attaching to thread 5
Attaching to thread 6
Attaching to thread 7
Attaching to thread 8
Attaching to thread 9
Attaching to thread 10
Attaching to thread 11
Attaching to thread 12
Attaching to thread 13
[J] Attaching to thread 14
Attaching to thread 15
## Attaching to thread 16
## Attaching to thread 17
## Attaching to thread 18
## Attaching to thread 19
## Attaching to thread 20
## Attaching to thread 21
## Attaching to thread 22
## Attaching to thread 23
## Attaching to thread 24
## Attaching to thread 25
## Attaching to thread 26
## Attaching to thread 27
[J] Attaching to thread 28
## Attaching to thread 29
## Attaching to thread 30
## Attaching to thread 31
[J] >>Sum of the array: 500010842.657334
[J] >>Total time to calculate the sum of the array: 8.026845 seconds
## =====
[J]
[J] Standard Deviation of the Array(parallelized)
Po
[J] Arguments: 3125000 0.000000 0 5.000108
/d Arguments: 3125000 0.000000 31250000 5.000108
de Arguments: 3125000 0.000000 62500000 5.000108
/d Arguments: 3125000 0.000000 93750000 5.000108
## Arguments: 3125000 -nan 12500000 5.000108
## Arguments: 3125000 0.000000 15625000 5.000108
[J] Arguments: 3125000 0.000000 18750000 5.000108
[J] Arguments: 3125000 0.000000 21875000 5.000108
[J] Arguments: 3125000 0.000000 25000000 5.000108
[J] Arguments: 3125000 0.000000 28125000 5.000108
[J] Arguments: 3125000 0.000000 31250000 5.000108
Arguments: 3125000 0.000000 34375000 5.000108
Arguments: 3125000 0.000000 37500000 5.000108
Arguments: 3125000 0.000000 40625000 5.000108
Arguments: 3125000 0.000000 43750000 5.000108
Arguments: 3125000 0.000000 46875000 5.000108
Arguments: 3125000 0.000000 50000000 5.000108
Arguments: 3125000 0.000000 53125000 5.000108
Arguments: 3125000 0.000000 56250000 5.000108
Arguments: 3125000 0.000000 59375000 5.000108
Arguments: 3125000 0.000000 62500000 5.000108
Arguments: 3125000 0.000000 65625000 5.000108
Arguments: 3125000 0.000000 68750000 5.000108
```

```
Terminal Shell Edit View Window Help
sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Arguments: 3125000 0.000000 9375000 5.000108
To Arguments: 3125000 -nan 12500000 5.000108
Arguments: 3125000 0.000000 15625000 5.000108
== Arguments: 3125000 0.000000 18750000 5.000108
== Arguments: 3125000 0.000000 21875000 5.000108
## Arguments: 3125000 -nan 25000000 5.000108
## Arguments: 3125000 0.000000 28125000 5.000108
## Arguments: 3125000 0.000000 31250000 5.000108
## Arguments: 3125000 0.000000 34375000 5.000108
## Arguments: 3125000 0.000000 37500000 5.000108
## Arguments: 3125000 0.000000 40625000 5.000108
## Arguments: 3125000 0.000000 43750000 5.000108
## Arguments: 3125000 0.000000 46875000 5.000108
## Arguments: 3125000 0.000000 50000000 5.000108
[D] Arguments: 3125000 0.000000 53125000 5.000108
[J] Arguments: 3125000 0.000000 56250000 5.000108
## Arguments: 3125000 0.000000 62500000 5.000108
## Arguments: 3125000 0.000000 65625000 5.000108
[D] Arguments: 3125000 0.000000 68750000 5.000108
[J] Arguments: 3125000 0.000000 71875000 5.000108
[J] Arguments: 3125000 0.000000 75000000 5.000108
no Arguments: 3125000 0.000000 78125000 5.000108
[J] Arguments: 3125000 0.000000 81250000 5.000108
[J] Arguments: 3125000 0.000000 84375000 5.000108
[J] Arguments: 3125000 0.000000 87500000 5.000108
Po Arguments: 3125000 0.000000 90625000 5.000108
[J] Arguments: 3125000 0.000000 93750000 5.000108
[d] Arguments: 3125000 0.000000 96875000 5.000108
de Attaching to thread 0
/d Attaching to thread 1
ma Attaching to thread 2
ma Attaching to thread 3
jl Attaching to thread 4
jl Attaching to thread 5
jl Attaching to thread 6
jl Attaching to thread 7
jl Attaching to thread 8
jl Attaching to thread 9
Attaching to thread 10
Attaching to thread 11
Attaching to thread 12
Attaching to thread 13
Attaching to thread 14
Attaching to thread 15
Attaching to thread 16
Attaching to thread 17
Attaching to thread 18
Attaching to thread 19
Attaching to thread 20
Attaching to thread 21
Attaching to thread 22
Attaching to thread 23
Attaching to thread 24
/d Attaching to thread 25
de Attaching to thread 26
/d Attaching to thread 27
ma Attaching to thread 28
ma Attaching to thread 29
jl Attaching to thread 30
jl Attaching to thread 31
[J]
[J] >>Standard deviation of parallelized array: 2.886603
[J] >>Total time to calculate parallelized standard deviation: 0.401223 seconds
>>Standard deviation of original/serial array: 2.886603
>>Result of original/serial standard deviation (i.e., HV1): 2.886603
>>Total time to calculate original/serial standard deviation: 1.809964 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 0.000285 seconds
[jlland@qbert PosixProgram]$
```

```
Terminal Shell Edit View Window Help
sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Arguments: 3125000 0.000000 90625000 5.000108
To Arguments: 3125000 0.000000 93750000 5.000108
Arguments: 3125000 0.000000 96875000 5.000108
== Attaching to thread 0
== Attaching to thread 1
## Attaching to thread 2
## Attaching to thread 3
## Attaching to thread 4
## Attaching to thread 5
## Attaching to thread 6
## Attaching to thread 7
## Attaching to thread 8
## Attaching to thread 9
## Attaching to thread 10
[D] Attaching to thread 11
[J] Attaching to thread 12
-- Attaching to thread 13
## Attaching to thread 14
-- Attaching to thread 15
[D] Attaching to thread 16
[J] Attaching to thread 17
[J] Attaching to thread 18
no Attaching to thread 19
[J] Attaching to thread 20
[J] Attaching to thread 21
[J] Attaching to thread 22
Po Attaching to thread 23
[J] Attaching to thread 24
/d Attaching to thread 25
de Attaching to thread 26
/d Attaching to thread 27
ma Attaching to thread 28
ma Attaching to thread 29
jl Attaching to thread 30
jl Attaching to thread 31
[J]
[J] >>Standard deviation of parallelized array: 2.886603
[J] >>Total time to calculate parallelized standard deviation: 0.401223 seconds
>>Standard deviation of original/serial array: 2.886603
>>Result of original/serial standard deviation (i.e., HV1): 2.886603
>>Total time to calculate original/serial standard deviation: 1.809964 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 0.000285 seconds
[jlland@qbert PosixProgram]$
```

Terminal Shell Edit View Window Help

sland — jlaland@qbert:~/PosixProgram — ssh jlaland@qbert.utc.edu — 160x50

```
Homework 2 Synopsis:  
Yo =====  
## Total number of threads: 64  
## Size of array: 100000000  
## Weight of smooth: 0.100000  
## =====  
## Creating the Array  
## Arguments: 15625000 10.000000 0  
## Arguments: 15625000 10.000000 15625000  
## Arguments: 15625000 10.000000 31250000  
## Arguments: 15625000 10.000000 46875000  
[J] Arguments: 15625000 10.000000 62500000  
## Arguments: 15625000 10.000000 78125000  
## Arguments: 15625000 10.000000 93750000  
## Arguments: 15625000 10.000000 109375000  
## Arguments: 15625000 10.000000 125000000  
[J] Arguments: 15625000 10.000000 140625000  
[J] Arguments: 15625000 10.000000 156250000  
no Arguments: 15625000 10.000000 171875000  
[J] Arguments: 15625000 10.000000 187500000  
[J] Arguments: 15625000 10.000000 203125000  
[J] Arguments: 15625000 10.000000 218750000  
Po Arguments: 15625000 10.000000 234375000  
[J] Arguments: 15625000 10.000000 250000000  
/d Arguments: 15625000 10.000000 265625000  
de Arguments: 15625000 10.000000 281250000  
/d Arguments: 15625000 10.000000 296875000  
ma Arguments: 15625000 10.000000 312500000  
## Arguments: 15625000 10.000000 328125000  
[J] Arguments: 15625000 10.000000 343750000  
[J] Arguments: 15625000 10.000000 359375000  
[J] Arguments: 15625000 10.000000 375000000  
[J] Arguments: 15625000 10.000000 390625000  
[J] Arguments: 15625000 10.000000 406250000  
Arguments: 15625000 10.000000 421875000  
Arguments: 15625000 10.000000 437500000  
Arguments: 15625000 10.000000 453125000  
Arguments: 15625000 10.000000 468750000  
Arguments: 15625000 10.000000 484375000  
Arguments: 15625000 10.000000 500000000  
Arguments: 15625000 10.000000 515625000  
Arguments: 15625000 10.000000 531250000  
Arguments: 15625000 10.000000 546875000  
Arguments: 15625000 10.000000 562500000  
Arguments: 15625000 10.000000 578125000  
Arguments: 15625000 10.000000 593750000
```

Terminal Shell Edit View Window Help

sland — jlaland@qbert:~/PosixProgram — ssh jlaland@qbert.utc.edu — 160x50

```
Yo Arguments: 15625000 10.000000 578125000  
[J] Arguments: 15625000 10.000000 593750000  
Arguments: 15625000 10.000000 609375000  
## Arguments: 15625000 10.000000 625000000  
## Arguments: 15625000 10.000000 640625000  
## Arguments: 15625000 10.000000 656250000  
## Arguments: 15625000 10.000000 671875000  
## Arguments: 15625000 10.000000 687500000  
## Arguments: 15625000 10.000000 703125000  
## Arguments: 15625000 10.000000 718750000  
## Arguments: 15625000 10.000000 734375000  
## Arguments: 15625000 10.000000 750000000  
## Arguments: 15625000 10.000000 765625000  
## Arguments: 15625000 10.000000 781250000  
## Arguments: 15625000 10.000000 796875000  
[J] Arguments: 15625000 10.000000 812500000  
## Arguments: 15625000 10.000000 828125000  
## Arguments: 15625000 10.000000 843750000  
## Arguments: 15625000 10.000000 859375000  
## Arguments: 15625000 10.000000 875000000  
[J] Arguments: 15625000 10.000000 890625000  
[J] Arguments: 15625000 10.000000 906250000  
no Arguments: 15625000 10.000000 921875000  
[J] Arguments: 15625000 10.000000 937500000  
[J] Arguments: 15625000 10.000000 953125000  
[J] Arguments: 15625000 10.000000 968750000  
Po Arguments: 15625000 10.000000 984375000  
[J] Attaching to thread 0  
/d Attaching to thread 1  
de Attaching to thread 2  
/d Attaching to thread 3  
ma Attaching to thread 4  
ma Attaching to thread 5  
[J] Attaching to thread 6  
[J] Attaching to thread 7  
[J] Attaching to thread 8  
[J] Attaching to thread 9  
[J] Attaching to thread 10  
Attaching to thread 11  
Attaching to thread 12  
Attaching to thread 13  
Attaching to thread 14  
Attaching to thread 15  
Attaching to thread 16  
Attaching to thread 17  
Attaching to thread 18  
Attaching to thread 19  
Attaching to thread 20  
Attaching to thread 21  
Attaching to thread 22
```

```
Terminal Shell Edit View Window Help sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Attaching to thread 32
Yo Attaching to thread 33
Attaching to thread 34
== Attaching to thread 35
== Attaching to thread 36
## Attaching to thread 37
== Attaching to thread 38
== Attaching to thread 39
== Attaching to thread 40
== Attaching to thread 41
== Attaching to thread 42
== Attaching to thread 43
== Attaching to thread 44
== Attaching to thread 45
D Attaching to thread 46
[J] Attaching to thread 47
== Attaching to thread 48
## Attaching to thread 49
== Attaching to thread 50
D Attaching to thread 51
[J] Attaching to thread 52
[J] Attaching to thread 53
mo Attaching to thread 54
[J] Attaching to thread 55
[J] Attaching to thread 56
[J] Attaching to thread 57
Po Attaching to thread 58
[J] Attaching to thread 59
/d Attaching to thread 60
de Attaching to thread 61
/d Attaching to thread 62
ma Attaching to thread 63
ma
jl >>Total time to create array: 14.402203 seconds
[J]
[J] ****
[J] Summing the Array
Arguments: 1562500 8
Arguments: 1562500 1562500
Arguments: 1562500 3125000
Arguments: 1562500 4687500
Arguments: 1562500 6250000
Arguments: 1562500 7812500
Arguments: 1562500 9375000
Arguments: 1562500 10937500
Arguments: 1562500 12500000
Arguments: 1562500 14062500
Arguments: 1562500 15625000
[J]
```

```
Terminal Shell Edit View Window Help sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Arguments: 1562500 5937500
Yo Arguments: 1562500 60937500
Arguments: 1562500 62500000
== Arguments: 1562500 64062500
== Arguments: 1562500 65625000
## Arguments: 1562500 67187500
== Arguments: 1562500 68750000
== Arguments: 1562500 70312500
== Arguments: 1562500 71875000
== Arguments: 1562500 73437500
== Arguments: 1562500 75000000
== Arguments: 1562500 76562500
== Arguments: 1562500 78125000
D Arguments: 1562500 79687500
D Arguments: 1562500 81250000
[J] Arguments: 1562500 82812500
== Arguments: 1562500 84375000
## Arguments: 1562500 85937500
== Arguments: 1562500 87500000
D Arguments: 1562500 89062500
[J] Arguments: 1562500 90625000
[J] Arguments: 1562500 92187500
mo Arguments: 1562500 93750000
[J] Arguments: 1562500 95312500
[J] Arguments: 1562500 96875000
[J] Arguments: 1562500 98437500
Po Attaching to thread 0
[J] Attaching to thread 1
/d Attaching to thread 2
de Attaching to thread 3
/d Attaching to thread 4
ma Attaching to thread 5
ma Attaching to thread 6
jl Attaching to thread 7
[J] Attaching to thread 8
[J] Attaching to thread 9
[J] Attaching to thread 10
[J] Attaching to thread 11
Attaching to thread 12
Attaching to thread 13
Attaching to thread 14
Attaching to thread 15
Attaching to thread 16
Attaching to thread 17
Attaching to thread 18
Attaching to thread 19
Attaching to thread 20
Attaching to thread 21
Attaching to thread 22
Attaching to thread 23
```

```
Terminal Shell Edit View Window Help
sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Attaching to thread 49
Yo Attaching to thread 50
Attaching to thread 51
== Attaching to thread 52
== Attaching to thread 53
## Attaching to thread 54
== Attaching to thread 55
== Attaching to thread 56
== Attaching to thread 57
== Attaching to thread 58
== Attaching to thread 59
== Attaching to thread 60
== Attaching to thread 61
== Attaching to thread 62
== Attaching to thread 63
[J1] == >>Sum of the array: 500010842.857344
## >>Total time to calculate the sum of the array: 0.025149 seconds
[J1]
=====
[J1] Standard Deviation of the Array(parallelized)
mo
[J1] Arguments: 15625000 0.000000 0 5.000108
[J1] Arguments: 15625000 0.000000 15625000 5.000108
[J1] Arguments: 15625000 0.000000 31250000 5.000108
Po Arguments: 15625000 0.000000 46875000 5.000108
[J1] Arguments: 15625000 0.000000 62500000 5.000108
/d Arguments: 15625000 0.000000 78125000 5.000108
de Arguments: 15625000 0.000000 93750000 5.000108
/d Arguments: 15625000 0.000000 109375000 5.000108
ma Arguments: 15625000 0.000000 125000000 5.000108
ma Arguments: 15625000 0.000000 140625000 5.000108
[J1] Arguments: 15625000 0.000000 156250000 5.000108
[J1] Arguments: 15625000 0.000000 171875000 5.000108
[J1] Arguments: 15625000 0.000000 187500000 5.000108
[J1] Arguments: 15625000 0.000000 203125000 5.000108
[J1] Arguments: 15625000 0.000000 218750000 5.000108
Arguments: 15625000 0.000000 234375000 5.000108
Arguments: 15625000 0.000000 250000000 5.000108
Arguments: 15625000 0.000000 265625000 5.000108
Arguments: 15625000 0.000000 281250000 5.000108
Arguments: 15625000 0.000000 296875000 5.000108
Arguments: 15625000 0.000000 312500000 5.000108
Arguments: 15625000 0.000000 328125000 5.000108
Arguments: 15625000 0.000000 343750000 5.000108
Arguments: 15625000 0.000000 359375000 5.000108
Arguments: 15625000 0.000000 375000000 5.000108
Arguments: 15625000 0.000000 390625000 5.000108
Arguments: 15625000 0.000000 406250000 5.000108
[FUSE]
[FUSE Volume Screen Shot 18-02...7.40 PM]
[ADS]
[ADS Screen Shot 18-02...7.47 PM]
[analysis]
[analysis of Screen Shot 18-02...7.50 PM]
[Big O]
[Big O Screen Shot 18-02...7.53 PM]
[skyAssignment]
[skyAssignment Screen Shot 18-02...7.54 PM]
```

```
Terminal Shell Edit View Window Help
sland — jlland@qbert:~/PosixProgram — ssh jlland@qbert.utc.edu — 160x50
Attaching to thread 4
Yo Attaching to thread 5
Attaching to thread 6
== Attaching to thread 7
== Attaching to thread 8
## Attaching to thread 9
== Attaching to thread 10
== Attaching to thread 11
== Attaching to thread 12
== Attaching to thread 13
== Attaching to thread 14
== Attaching to thread 15
== Attaching to thread 16
== Attaching to thread 17
[J1] Attaching to thread 18
[J1] Attaching to thread 19
== Attaching to thread 20
## Attaching to thread 21
== Attaching to thread 22
[J1] Attaching to thread 23
[J1] Attaching to thread 24
[J1] Attaching to thread 25
mo Attaching to thread 26
[J1] Attaching to thread 27
[J1] Attaching to thread 28
[J1] Attaching to thread 29
Po Attaching to thread 30
[J1] Attaching to thread 31
/d Attaching to thread 32
de Attaching to thread 33
/d Attaching to thread 34
ma Attaching to thread 35
ma Attaching to thread 36
[J1] Attaching to thread 37
[J1] Attaching to thread 38
[J1] Attaching to thread 39
[J1] Attaching to thread 40
[J1] Attaching to thread 41
Attaching to thread 42
Attaching to thread 43
Attaching to thread 44
Attaching to thread 45
Attaching to thread 46
Attaching to thread 47
Attaching to thread 48
Attaching to thread 49
Attaching to thread 50
Attaching to thread 51
Attaching to thread 52
Attaching to thread 53
[FUSE]
[FUSE Volume Screen Shot 18-02...7.40 PM]
[ADS]
[ADS Screen Shot 18-02...7.47 PM]
[analysis]
[analysis of Screen Shot 18-02...7.50 PM]
[Big O]
[Big O Screen Shot 18-02...7.53 PM]
[skyAssignment]
[skyAssignment Screen Shot 18-02...7.54 PM]
```

Terminal Shell Edit View Window Help

jland — jland@qbert:~/PosixProgram — ssh jland@qbert.utc.edu — 160x50

```
Attaching to thread 29
Yo Attaching to thread 30
Attaching to thread 31
== Attaching to thread 32
== Attaching to thread 33
## Attaching to thread 34
== Attaching to thread 35
== Attaching to thread 36
== Attaching to thread 37
== Attaching to thread 38
== Attaching to thread 39
== Attaching to thread 40
== Attaching to thread 41
== Attaching to thread 42
D Attaching to thread 43
D Attaching to thread 44
[J] Attaching to thread 45
== Attaching to thread 46
## Attaching to thread 47
D Attaching to thread 48
[J] Attaching to thread 49
[J] Attaching to thread 50
mo Attaching to thread 51
[J] Attaching to thread 52
[J] Attaching to thread 53
[J] Attaching to thread 54
Po Attaching to thread 55
[J] Attaching to thread 56
/d Attaching to thread 57
de Attaching to thread 58
/d Attaching to thread 59
ma Attaching to thread 60
ma Attaching to thread 61
jl Attaching to thread 62
[J] Attaching to thread 63
[J]
[J] >>Standard deviation of parallelized array: 2.886603
[J] >>Total time to calculate parallelized standard deviation: 0.306584 seconds
>>Standard deviation of original/serial array: 2.886603
>>Result of original/serial standard deviation (i.e., HV1): 2.886603
>>Total time to calculate original/serial standard deviation: 1.823138 seconds
=====
Smoothing the Array
Total time to calculate parallelized array after smooth: 0.076065 seconds
[jland@qbert PosixProgram]$
```