Comparative Analysis of Multi-Threaded and Non-Threaded

Implementations of the Merge Sort Algorithm

(Final Report)

By

Bernice Templeman

Florida Institute of Technology

CIS 5898 – Projects in Computer Information Systems

For

Dr. Stephen Johnson

November 7, 2016

# Table of Contents

Introduction 3

Sorting Decisions 3

Java Threads 3

BTMS Proposition 3

Algorithms / Project Solution 4

Merge-Sort ALgorithm 4

Threads 4

Data Set Size 4

Timing 4

Implementation 5

Platform 5

Development Tools 5

Programming Language and Code Libraries 5

Implementation Procedure 5

Schedule 6

Results 9

Future Enhancements 11

References 12

Appendices 13

# Introduction

Bernice’s Threaded Merge Sort (BTMS) will sort large datasets with Java using the merge sort algorithm.

**Student Learning Outcomes (SLOs):** The overall outcome is for the adult learner to support professional performance with a solid foundation in knowledge. Students demonstrate competency in data structures, computer programming, research in the computer sciences, computer architecture, and operating systems. This is accomplished through:

1. Applying the techniques and principles that underlie information systems design and implementation.
2. Applying the techniques and principles that underlie component and object oriented design.
3. Demonstrating an understanding of the role of information systems within small, medium and large scale network systems.
4. Develop and propose a significant IT/IS problem and its solution (the capstone project).

**Project Goals:**

1. To demonstrate the ability to do academic research in the field of information systems.
2. To demonstrate capable use of a toolset which includes data structures, object-oriented design and programming, computer architecture, and operating systems.
3. Write a proposal and have it accepted by the course instructor.
4. Implement the proposed system or evaluation or extension of the systems proposed.
5. Supply 5 or more progress reports reflecting the progress of the project.

**Part 1. Purpose of the Capstone Course**

For each MS/CIS student, this required capstone course is a graduation requirement. It provides an avenue for the student to show how they can apply the course knowledge obtained in Florida Tech’s Master of Science in Computer Information Systems curriculum. A capstone project can be centered around any of a number of concentrations including, but not limited to the list below. It is even better to merge 2 or more.

In general, a project is expected to implement a system or sub-system of software components (developed for either Windows or Linux OS). Re-usable components taken off the World Wide Web can be used to build larger systems as long as credit is given to those implementers where it is due. The project should also include a certain amount of research comparable to a mini-thesis.

Phase I. Research and Proposal Writing Phase

Phase II. Design and Implementation Phase (Deliverable: Design and Code, Screen Capture Video [ in zipped format], Virtual Machine appliance [i.e., export of applications entire environment in an open format, \*.ova file ] )

Phase III. Report Writing (Deliverable: Final *Report [NOTE: This Report merges with PhaseI, and documentation from the Design and Code Deliverable of Phase II.]*)

Students generate a report on the project including what was done, how it was done, lessons learned, etc. A List of References or Bibliography should accompany the paper.

**Part 3. Possible Topic Areas**

**Part 4. Proposal**

**Part 5. Some sample ideas**

**Part 6. The report.**

**Final Project Submittal**

Class,

As we near the completion of the capstone I want to reiterate a few items. This is to mainly ensure that all required project artifacts are submitted in the correct format.

* Review the "Capstone Project Requirements" document for the format of the final report. In addition, please review the Capstone Project example located in the "Completed Capstone Project - Example" module in Canvas. Do not include the source code in the appendix of the report.
* Provide Camstudio video screen capture with sound (refer to "Phase 2. Design and Implementation Phase", pg. 2 of the "Capstone Project Requirements" document).
* Address the project research requirements. This is described in the "Future Enhancements" section.
* Address the "Software Configuration" & "User Manual" sections. The "Software Configuration" section provides the information to setup an environment to run the project application.

Please be sure to submit all the final project artifacts (Final Report, application files, etc.) to the "Final Project Submission Area" located in Canvas by midnight EST Sunday, November 20th. As an alternative, if necessary, send the application files, VMs (for non-web based applications), CamStudio Video via U.S. Mail (refer to "Primary Course Documents" section in Canvas for Mailing Address). This is a firm deadline and any projects received after this date may be subject to a late penalty, which may affect the final grade. Please be sure that any comments provided earlier in the semester (e.g., emails, replies in the Discussion Board area) have been incorporated into the final project documentation.

**Research**

### Sorting Decisions

There are several sorting algorithms to choose from. With an increase in the amount of data being created, implementing the best sort for the data will affect performance. George Heineman says in *Algorithms in a Nutshell* that “Because today’s computers are so much more powerful than the ones of 50 years ago, the size of the data sets being processed is now on the order of terabytes of information. Although you may not be called on to sort such huge data sets, you will likely need to sort large numbers of items.” Merge Sort is considered as a stable sort. It is also considered as well-suited for sorting data in secondary storage. (Heineman).

### Java Threads

To further improve the merge sort performance, multi-threaded sorts are another option. In *An Implementation of Sorting Algorithm Based on Java Multithread Technology*, Wang says

“With the spread of muticore computer, ordinary desktop computers have strongparallel processing ability. But using traditional serial sort algorithm cannot take full advantage of powerful parallel computing power of the computer. Therefore, how to reduce the recursion level and how to improve the traditional algorithms, made him able to adapt to the development of computer parallel technology, to increase the efficiency of traditional algorithms to a new level, is a worthy subject of study.”

Washington University has the merge sort and threaded merge sort java code available online. I decided to use this code in my performance tests.

**Virtual Machines**

Virtual Machines are used in production, development and testing environments. With the increasing Cloud Services, Virtual Machines are also increasing. According to VMware, one of the most popular virtiual machines, …

<http://www.vmware.com/solutions/virtualization.html>

Another popular Virtual machine \_\_\_\_ is Oracle VirtualBox is shareware. It can be used on different platforms including Windows and Linux. For these reasons, I decided to use VirtualBox to satisfy the Virtual Machine project requirement.

<http://www.infoworld.com/article/2615128/virtualization/review--vmware-workstation-9-vs--virtualbox-4-2.html>

**Operating Systems**

Windows

Ubuntu

Windows host

Decided not to boot to a Ubuntu USB for performance reasons.

**Data Sets**

There are also different APIs to create Data Sets. A common API is the Random API which is included in Java. Another one is StdRandom which I found on the Princeton website. It includes methods to create different data sets. I chose the Random and some of the methods in the StdRandom API to test the merge sorts.

**User Interface**

## Graphical user interfaces use threads and impact performance. For these reasons I decided not to use a GUI and run the application from NetBeans. A JAR file will be created so the application can be ran with out NetBeans.

**Camtasia**

# Deliverables:

# Packaging and Distributing Java Desktop Applications

* Export Virtual Machines

## BTMS Proposition

BTMS will test the performance of a non-threaded merge sort and a multi-threaded merge sort java program on a Windows 10 Virtual Machine and an Ubuntu 16.04 LTS Virtual Machine. BTMS will also be ran on the Oracle VirtualBox Machine Host Machine for comparison purposes.

# Algorithms / Project Solution

This section describes the unique problem and the proposed solution that will be known as BTMS.

**The Merge-Sort Algorithm**

Merge-Sort is an algorithm that divides its input list into two lists, sorts them, and merges the two sub-lists into a sorted version of the input list.

MERGESORT(A)

ifLENGTH(A) =< 1

then returnA

B <- MERGESORT(first half of A)

C <- MERGESORT(second half of A)

A <- MERGE(B, C)

returnA

**Threads**

The multi-threaded program will allow the following number of threads to be used for the merge-sort:

* 1
* 2
* 4
* 8
* 16

Threads were only tested up to 16 threads. Due to hardware limitations (processors), increasing the number of threads to larger sizes did not improve performance.

**Data Set Size**

Different data set sizes will be created and tested. An array of integers will be created using the dataset. The following data set sizes were tested:

* 1000
* 2000
* 4000
* 8,000
* 16,000
* 32,000
* 64,000
* 128,000
* 256,000
* 512,000
* 1,024,000
* 2,048,000
* 4,096,000
* 8,192,000
* 16,384,000
* 32,768,000
* 65,536,000

Larger data sets were not tested due to hardware limitations.

**Data Set Type**

The following data sets were tested:

* Random
* Discrete
* Uniform
* Bernoulli
* Gaussian
* Zero

**Using Exiting Code and Libraries**

* **Merge Sort**
* **Threaded Merge Sort**
* **Random**
* **STDRandom**

**Hardware**

**Operating Systems**

* **Windows 10**
* **Ubuntu**

**Virtual Machines**

* **Oracle VirtualBox**

**Timing**

The time will be taken before and after the sort is performed using: System.currentTimeMillis();

**Anti-Virus**

* **McAfee**

**Configuration Management**

* **Github**
* **Google drive**

**Reports**

* **Word**
* **Excel**
* **Camtasia**
* **Gantt**

# Implementation

This section describes the resources that will be used to implement BTMS.

## Platform

BTMS will be a Java 8 application running. It will be installed on an Intel platform running:

* Windows 10 Operating System (OS)
* Oracle Virtual Box Windows 10 VM
* Oracle VM Virtual Box 5..1.8 ( with extension pack)Ubuntu VM

Performance will not be tested on an Ubuntu bootable USB drive because research found shows there are performance costs running on an USB. <http://askubuntu.com/questions/21741/performance-cost-of-running-ubuntu-from-external-hard-drive>

Information and downloads for Java can be found at <https://docs.oracle.com/javase/8/>.

## Development Tools

Development will be done on an Intel machine running Windows 10 and a Linux OS Bootable drive. I will be using NetBeans IDE for Java EE Developers. Information and downloads can be found at <http://www.oracle.com/technetwork/articles/javase/jdk-netbeans-jsp-142931.html>.

## Programming Language and Code Libraries

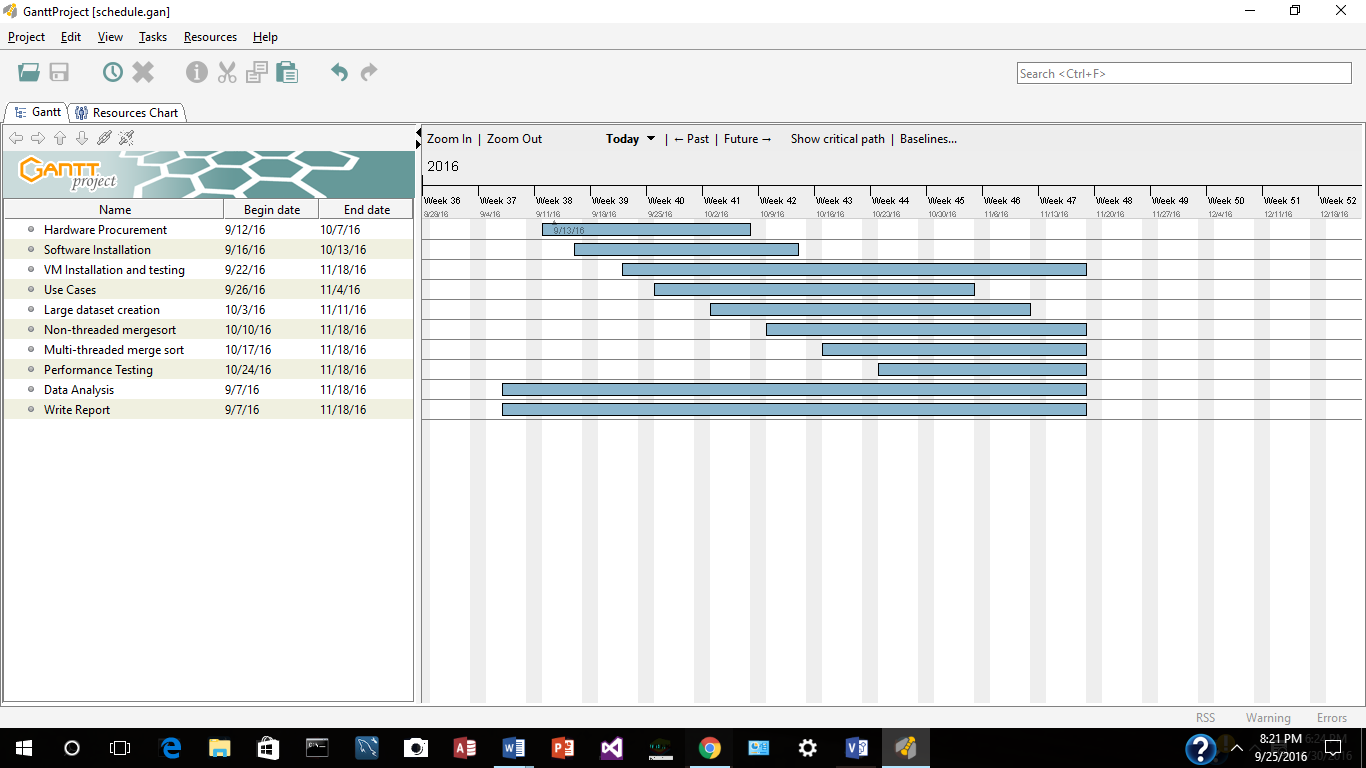
BTMS will be programmed in the Java programming language using the NetBeans IDE.

## Implementation Procedure

I plan to start by procuring a Windows 10 laptop, installing OS updates and installing software. I will also install Oracle VirtualBox, download a Windows 10 VM and test it during this phase. Next I will develop high-level use case diagrams for the threaded application. Also during this phase I’ll design the interface. Next, I will install Java and the NetBeans IDE on both Windows 10 hard and an Ubuntu Linux 14.04. I will research existing datasets, code to produce a dataset or writing a Java program to create a large dataset of integers to be sorted by the merge-sort programs. I will then write a non-threaded merge-sort. After the non-threaded merge-sort is complete I will write the multi-threaded merge-sort. Then I will run performance tests to compare the non-threaded merge-sort with the threaded merge-sort with different number of threads.

|  |
| --- |
| Model Samsung - Notebook 7 Spin 2-in-1 15.6" |
| **Platform A** Windows 7 Enterprise  Edition |
| **Platform B** Windows 8.1 Pro |
| **Platform C** Ubuntu 14.04 LTS |
| **CPU** 2.66GHz x 4, 8 MB Cache, |
| **Processor** Intel(R) Core(TM) i7-6500U CPU @ 2.50GHz, 2592 Mhz, 2 Core(s), 4 Logical Processor(s)  **Model** |
| **Hard Disk** 1TB Hard Drive |
| **RAM** 12 GB |
| **Graphics** NVIDIA GeForce 940MX |
| Fig. 3 Hardware Platform Used in  Experiments |

# Schedule



# Results

**Student Learning Outcomes (SLOs):** The overall outcome is for the adult learner to support professional performance with a solid foundation in knowledge. Students demonstrate competency in data structures, computer programming, research in the computer sciences, computer architecture, and operating systems. This is accomplished through:

1. Applying the techniques and principles that underlie information systems design and implementation.
2. Applying the techniques and principles that underlie component and object oriented design.
3. Demonstrating an understanding of the role of information systems within small, medium and large scale network systems.
4. Develop and propose a significant IT/IS problem and its solution (the capstone project).

**Project Goals:**

1. To demonstrate the ability to do academic research in the field of information systems.
2. To demonstrate capable use of a toolset which includes data structures, object-oriented design and programming, computer architecture, and operating systems.
3. Write a proposal and have it accepted by the course instructor.
4. Implement the proposed system or evaluation or extension of the systems proposed.
5. Supply 5 or more progress reports reflecting the progress of the project.

**Deliverables (see “Capstone Project Requirements” document for details):**

Progress Reports

Final Report

Project Submission and Video Screen Capture

. The “Project Report” and related software application files must be posted to the “Final Project Drop Box”.

**Part 1. Purpose of the Capstone Course**

For each MS/CIS student, this required capstone course is a graduation requirement. It provides an avenue for the student to show how they can apply the course knowledge obtained in Florida Tech’s Master of Science in Computer Information Systems curriculum. A capstone project can be centered around any of a number of concentrations including, but not limited to the list below. It is even better to merge 2 or more.

In general, a project is expected to implement a system or sub-system of software components (developed for either Windows or Linux OS). Re-usable components taken off the World Wide Web can be used to build larger systems as long as credit is given to those implementers where it is due. The project should also include a certain amount of research comparable to a mini-thesis.

**Part 2. Structure of the Course**

The general structure of the course should include the following phases and deliverables:

Phase I. Research and Proposal Writing Phase (Deliverable: Proposal)

During this phase students should research some of the topics (see Part 3) and should come up with a project. Then, the student is expected to submit a proposal for the work to be done for the Capstone experience. Proposals must be submitted to the CIS 5080 Instructor and approved. There may need to be some negotiation regarding proposals in order to establish exactly what the sort of project a student will do and what the deliverables are. The proposal should be handed in and approved around the end of the third week of classes, if not earlier.

Phase II. Design and Implementation Phase (Deliverable: Design and Code, Screen Capture Video [ in zipped format], Virtual Machine appliance [i.e., export of applications entire environment in an open format, \*.ova file ] )

Once the course instructor approves the proposal submitted by the student, the student researches, designs, and implements the project. The code that will be delivered, as part of this phase, will be tested by the instructor. If a Web application is submitted, the student must provide access to the completed application by providing the instructor with the URL for the hosted Web application. During this phase Progress Reports (4) are handed in outlining progress. Changes to the proposal during this phase should be documented and run by the instructor also. In addition to the submitted code, a video (5 - 10 min max.) of the running application must be provided. Freely available tools such as CamStudio, http://camstudio.org/ can be used to make video screen captures.

Phase III. Report Writing (Deliverable: Final *Report [NOTE: This Report merges with PhaseI, and documentation from the Design and Code Deliverable of Phase II.]*)

Students generate a report on the project including what was done, how it was done, lessons learned, etc. A List of References or Bibliography should accompany the paper.

**Part 4. Proposal**

II Introduction

In a nutshell, say what you're going to do and how you're going to do it. Provide justification as to why this is a worthy endeavor. Explain the practical applications of your project. This is an opportunity to sell your idea to management of a company. This section should provide as much supporting materials for your project as is practical. If strong justification cannot be provided here, pursue another topic.

III. Algorithms/Project Solution

This section should describe the resources that will be used to implement your project. Specifically, describe the well-known Algorithms that will be used. If there are no well-known algorithms that will be used, describe this unique problem and the proposed solution. The types of resources include (but are not limited to), design patterns, architectural patterns, generic algorithms, The information included in this section would be similar to what would be found in the Requirements, Analysis and Design phases of the Systems Development Life Cycle (SDLC).

IV. Implementation

This section will describe all the specific resources that will be used to implement the projects. These will include programming languages, application frameworks (e.g., .NET, J2EE, etc.) database management systems, compilers, operating systems and hardware platforms. All project code must be developed for one of the following operating systems.

* Linux
* Windows XP, Windows 2000, Windows 7

For vendor specific tools, provide the appropriate vendor information for that tool (i.e., manufacture name, website, etc.). For other tools, provide similar information (e.g., Apache Software Foundation). Describe the purpose of each tool and the integration of each tool to develop the complete solution for the project. Provide a high-level procedure that will be used to implement the project solution.

V. Schedule (Gantt Chart)

VI. References

This part is not optional. List all sources, including net sources here.

*NOTE: A carefully planned out project would be designed and crafted so that a large portion of the proposal could be re-used in the final report.*

Every project proposed must include the following components

* The proposal itself can be anywhere between 5 and 20 pages.
* Since all software pretty much can be defined as something that provides a specific function or solves a particular problem or set of problems, this is a key element of the proposal. Examples: Design of a Program that implements the game of Mastermind with the following additions; Implementation of the Round Robin Scheduler in Unix with Enhancements; Gaussian Elimination implemented in C++ and Language XYZ; a Performance Comparison.
* Every project must consist of as a minimum a) an accepted proposal, b) some software component or software system, and c) a report.
* The software component/system must be written by one individual, can be a hybrid of reusable components in different languages or using different tools (SQL, .NET, Thread Libraries, Java Libraries, scripts, etc.), must be implemented within a virtual machine (e.g., Oracle VirtualBox). In some cases the building of a larger system composed mainly of reusable software components would be appropriate. The following are acceptable software technologies.
  + Operating Systems - Windows or Linux
  + Database - SQL Server, MySQL, JavaDB
  + Programming Languages - C++, Java, C#, PHP, JavaScript
  + Development Environments - Visual Studio 2008/2010, Eclipse, NetBeans

Other Notes about the Proposals

* Think of it as similar to a Requirements Document.
* Expect that there may be negotiation on the proposal. This is due to the open nature of the topic areas.
* Projects can be work related, but just submitting something from work will not suffice.
* All projects must incorporate some form of computer software. Reusable component use is acceptable and encouraged as long as credit is given to where (website, text, paper, etc.) that the software was obtained from. “Invisible” classes or objects that are used don’t necessarily need to be documented but they can be. For instance, it would be unnecessary to document the stream classes you use in your Java system. But, if you use the Socket class quite a bit when designing a server/client application, you may find it quite helpful to document its use.
* It is expected that the student will do a certain amount of research to do this project. The level of research expected is nowhere near what is expected for a graduate level thesis, but since none of us live and think in a vacuum, it is always a good idea to support the ideas you propose and implement with texts and papers. The Florida Tech LINK is key here to providing tools for doing online research.
* Be careful with web based resources! Anybody can set up a web page and the contents of web pages usually aren’t put under the same scrutiny as published texts and papers. Understand this and make sure your Bibliographies contain more than just web page/URL references.
* If you find yourself choosing a topic area that is not publishable, don’t choose it.

**Part 5. Some sample ideas**

There are literally an infinite number of projects that can be proposed for the Capstone experience. Below is a list relating to Operating Systems/Thread Programming. Note: MP = Multi-process and MT = Multi-threaded.

* Take a well-known algorithm and implement it with MP's and MT's. Compare the performance of both implementations.
* Implement just about any algorithm (that makes sense) with MP's, and compare it to a single process solution.
* Likewise, implement just about any algorithm (that makes sense) with MT's, and compare it to a single process solution.
* Collect algorithms and results from others in the field. This means your code-writing is kept to a minimum. Emphasis here must be on the final report and the findings.
* Compare solutions to well known OS problem: a Java solution to a C or C++ solution.
* Do the above comparing language XYZ to ABC (fill in your own).
* Do any of the above in a single processor environment. Compare its performance to a multi-processor environment.
* Compare an MT or MP solution on a Unix platform and compare it to the same solution on a Windows platform.
* Compare an MT or MP solution on a Unix platform and compare it to the same solution on a Linux platform.

One could easily come up with many more types of algorithms for any of the disciplines listed in Part 1. Ask your Capstone instructor if you need ideas.

**Part 6. The report.**  This should be a well organized report of your findings and results. Reiterate what it was you were trying to do. The report is typically 15-30 pages without code. All reports should have a *List of References*. Students are expected to do a certain amount of research for the proposal and the final report. The format of the final report will be as follows.

I. Title Page similar to Proposal except use “Final Report” instead of “Proposal”

II. Table of Contents - Entries must be hyperlinked (i.e., clicking will advance to that section in the document)

III Introduction ***update from original proposal if necessary***

IV. Algorithms/Project Solution ***update from original proposal if necessary***

V. Implementation ***update from original proposal if necessary***

VI. Schedule (Gantt Chart) ***update from original proposal if necessary***

VII. Results

This section should describe your experience in implementing the solution. Describe any problems encountered, solutions implemented and reasons for choosing that solution when multiple alternatives were available. Describe any “lessons learned” that could be beneficial to future implementers.

In addition, if the project is a comparative analysis, this section should describe the results of the comparison. Use charts, graphs, and other means to summarize data if this would provide a clearer explanation of results. and whether the results were as predicted ahead of time. Provide very detailed explanation of the results.

VIII. Future Enhancements

This section should provide ideas for possible extensions of this project. **This section must describe an area of research (and reference at least one appropriate refereed journal article) that could be incorporated into this project**. For example, if this application was a database related project, perform research on search algorithms and describe how these could be implemented to enhance the functionality of this application. Another example for a database related project would be to research alternative architectures to relational (e.g., multidimensional, object-oriented, etc.).

IX. References

This part is not optional. List all sources, including net sources here (limit Web references to no more than 50%). **In addition there should be no “Wiki” references.**

X Appendix

Progress Reports

This section will be a copy of the progress reports that have been submitted for the project. The report will contain the following sections

* Date: *date of this report*
* Accomplishments: *what has been completed since last progress report*
* Current Activities: *what is currently being worked on*
* Challenges: *what areas are time consuming or causing problems*
* Work to be Completed: *what needs to be completed for the next project milestone*

Software Configuration

This section provides all steps necessary to install and configure application to run

User’s Manual

This section provides all instructions for user to run the application and perform all functionality once the application has been installed

**Part 8. Other issues.**

• All projects (i.e. proposals, reports, and presentations) must include a *List of References* or *Bibliography*.

• All code should have in it, as a comment, the author. All code "borrowed" from the net/web should be referenced in the *List of References*. If I get code which looks borrowed, but you don't mention it, I will search the web myself. If I find it, then this will be cited as plagiarism. See the policies handout and the FIT student handbook to see how plagiarism will be handled.

• Material borrowed from a source in the report and proposal should reference the *List of References* and can either be superscripted like this1, or referenced like this [1].

• Use double quotes (“ “) when you take material directly from a reference. If you rearrange and mix ideas or add your own, then you needn’t use quotes. In either case, reference it. For appropriate formats for listing your references in the "List of References”, see the reference librarian. There are several textual (and online) references that will tell you how to list references appropriately.

**Results of Merge Sort**

**Comparing Merge Sort Times on different Operating Systems:**

I am a long time Unix person. I first wrote C and C++ programs on Unix Systems and then also did Unix System Administration. Later I also did RedHat Linux System Administration. There were always questions on which OS developers should use. When I first ran the merge sort on the different Operating Systems, I was surprised by the results.

The following are the results of the first runs on a Windows 10 and Ubuntu Virtual Machine with 1 Processor (Random Data Set Averages):

|  |  |
| --- | --- |
| Windows 10 VM (1 Processor) | Ubuntu VM (1 Processor) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Array size  izeize | NT | 1T | 2T | 4T | 8T | 16T | NT | 1T | 2T | 4T | 8T | 16T |
| **1,000** | 0 | 14.1 | 26.5 | 50.1 | 104.6 | 214.5 | 0.4 | 9.6 | 8.9 | 17.4 | 29.2 | 68.3 |
| **2,000** | 3.2 | 14.2 | 23.5 | 55 | 98.6 | 213.8 | 2.1 | 9.6 | 8.5 | 18.2 | 31.8 | 68.1 |
| **4,000** | 0 | 17.3 | 31.1 | 52.9 | 112.6 | 219 | 0 | 8.4 | 10.8 | 18.1 | 30.3 | 61.5 |
| **8,000** | 0 | 10.6 | 23.9 | 54.8 | 104.7 | 218.6 | 0.2 | 7.7 | 9.4 | 17.2 | 40.6 | 86.3 |
| **16,000** | 0 | 14.3 | 25.1 | 53.1 | 104.6 | 221.7 | 0.5 | 9.6 | 10.2 | 16 | 40.2 | 78.7 |
| **32,000** | 0 | 14.1 | 26.3 | 57.9 | 112.6 | 217.3 | 1.2 | 9.5 | 9.3 | 17 | 36.1 | 74.3 |
| **64,000** | 1.5 | 15.5 | 34.7 | 53.3 | 113.8 | 210.9 | 2.6 | 11.1 | 10.3 | 18.4 | 62.1 | 156.8 |
| **128,000** | 4.7 | 18.5 | 36 | 62 | 107.9 | 224.9 | 7.1 | 12.1 | 13.6 | 21.2 | 57.7 | 151.9 |
| **25,6000** | 9.5 | 23.4 | 40.6 | 72.3 | 118.9 | 233 | 11 | 27.5 | 19.7 | 27.6 | 62.3 | 160.4 |
| **512,000** | 19.9 | 35 | 50 | 82.4 | 136.2 | 231.2 | 33 | 30.3 | 36.3 | 77.3 | 82.7 | 202.3 |
| **1,024,000** | 43.9 | 56.1 | 70.1 | 103 | 155.9 | 264.2 | 76.9 | 75.3 | 62.2 | 96.1 | 143.5 | 237 |
| **2,048,000** | 81.7 | 98.5 | 123.5 | 149.6 | 201.4 | 295.1 | 187.7 | 142.6 | 187.6 | 166.9 | 292.1 | 373.2 |
| **4096000** | 168.8 | 185.9 | 201.3 | 237.5 | 287.2 | 423.1 | 381.6 | 290.3 | 430.5 | 475 | 501.4 | 561.9 |
| **8192000** | 351.7 | 357.7 | 393.8 | 427.9 | 476.6 | 614 | 902.3 | 869.6 | 946.7 | 890 | 894.2 | 967.6 |
| **16384000** | 693.5 | 696.8 | 736.2 | 765.3 | 815.7 | 1000.2 | 1452.8 | 1592.8 | 1600.5 | 1650.1 | 1614.5 | 1640.8 |
| **32768000** | 1366.8 | 1389.1 | 1398.4 | 1465.8 | 1531.2 | 1660.9 | 2990 | 2992.8 | 3002.3 | 3228.9 | 3010.8 | 3144.2 |
| **65,536,000** | 2673.3 | 2901.4 | 2785.8 | 2777.8 | 3015.4 | 3070.7 | 5346.3 | 5341.1 | 5295.5 | 5489.2 | 5325.7 | 5376.5 |

The results show that on the Virtual Machines with 1 processor, Windows 10 Non-Threaded was the fastest.

I then changed the Virtual Machine Configurations to 2 Processors and these are the results from the Random Data Sets (Averages):

|  |  |
| --- | --- |
| Windows 10 VM  (2 Processors) | Ubuntu VM  (2 Processors) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Array size  izeize | NT | 1T | 2T | 4T | 8T | 16T | NT | 1T | 2T | 4T | 8T | 16T |
| **1,000** | 0 | 40.3 | 37.6 | 64.1 | 121.9 | 248.2 | .1 | 23 | 26.8 | 58.9 | 106.9 | 220.2 |
| **2,000** | 0 | 32.6 | 40.4 | 67.1 | 124.7 | 250 | 1.3 | 17.8 | 22.8 | 52.8 | 107.9 | 225.6 |
| **4,000** | 1.5 | 28.2 | 40.4 | 64.2 | 122.2 | 256.5 | 0.1 | 17.9 | 25.8 | 49.5 | 119.8 | 232 |
| **8,000** | 0 | 31.3 | 37.6 | 67.2 | 132.5 | 248.1 | 0.1 | 23.3 | 25.9 | 72.1 | 124.1 | 225.1 |
| **16,000** | 3 | 27.8 | 39.1 | 63.9 | 120.3 | 251.6 | 1.5 | 17.8 | 25.5 | 60 | 118.3 | 206.2 |
| **32,000** | 1.5 | 26.8 | 35.8 | 61.3 | 125.2 | 254.6 | 1.3 | 17.5 | 29.9 | 57.3 | 121 | 228.7 |
| **64,000** | 0 | 26.6 | 37.5 | 60.8 | 117.1 | 262.6 | 2.9 | 19.3 | 33.3 | 72.5 | 127.2 | 236.9 |
| **128,000** | 3.1 | 35.7 | 39 | 65.4 | 126.6 | 252.8 | 5.1 | 29.9 | 37.2 | 76.3 | 121.1 | 287.1 |
| **25,6000** | 7.9 | 36 | 46.8 | 68.8 | 129.6 | 245.5 | 10.9 | 35.5 | 52.5 | 85.1 | 125.5 | 245.2 |
| **512,000** | 22 | 48.3 | 56.2 | 73.3 | 131.2 | 245.1 | 25.1 | 55.7 | 59.6 | 120 | 154.6 | 267.1 |
| **1,024,000** | 42.3 | 77.9 | 66.9 | 86 | 140.4 | 262.4 | 50.9 | 84.5 | 108 | 146.7 | 245.6 | 327.5 |
| **2,048,000** | 85.6 | 120.4 | 98.4 | 107.5 | 159.2 | 276.1 | 112.1 | 159.2 | 185.5 | 212.5 | 280.7 | 391.6 |
| **4096000** | 168.8 | 215.6 | 163.6 | 162.5 | 210.9 | 318.8 | 235.1 | 264.4 | 278.2 | 325.4 | 403.7 | 531.7 |
| **8192000** | 336.1 | 395.8 | 295.2 | 265.6 | 314.1 | 415.5 | 390.8 | 478.6 | 447.1 | 476.8 | 571 | 764.1 |
| **16384000** | 688.9 | 767.4 | 535.7 | 493.8 | 512.4 | 628.1 | 747.5 | 915.8 | 727.8 | 736.9 | 875.9 | 1071.1 |
| **32768000** | 1361 | 1429.6 | 1006.2 | 867.3 | 909.3 | 1017.3 | 1479.6 | 1674.4 | 1324.4 | 1325 | 1403.4 | 1544.6 |
| **65,536,000** | 2739 | 2815.1 | 1954.7 | 1657.6 | 1715.6 | 1800 | 2952.8 | 3255.5 | 2500.2 | 2538.2 | 2575.2 | 2725.4 |

**Observations Of adding a processor to Virtual Machine:**

**Random Data Set Averages**

|  |  |
| --- | --- |
| Windows 10 VM  (1 Processor) | Windows 10 VM  (2 Processors) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Array size  izeize | NT | 1T | 2T | 4T | 8T | 16T | NT | 1T | 2T | 4T | 8T | 16T |
| **1,000** | 0 | 14.1 | 26.5 | 50.1 | 104.6 | 214.5 | 0 | 40.3 | 37.6 | 64.1 | 121.9 | 248.2 |
| **2,000** | 3.2 | 14.2 | 23.5 | 55 | 98.6 | 213.8 | 0 | 32.6 | 40.4 | 67.1 | 124.7 | 250 |
| **4,000** | 0 | 17.3 | 31.1 | 52.9 | 112.6 | 219 | 1.5 | 28.2 | 40.4 | 64.2 | 122.2 | 256.5 |
| **8,000** | 0 | 10.6 | 23.9 | 54.8 | 104.7 | 218.6 | 0 | 31.3 | 37.6 | 67.2 | 132.5 | 248.1 |
| **16,000** | 0 | 14.3 | 25.1 | 53.1 | 104.6 | 221.7 | 3 | 27.8 | 39.1 | 63.9 | 120.3 | 251.6 |
| **32,000** | 0 | 14.1 | 26.3 | 57.9 | 112.6 | 217.3 | 1.5 | 26.8 | 35.8 | 61.3 | 125.2 | 254.6 |
| **64,000** | 1.5 | 15.5 | 34.7 | 53.3 | 113.8 | 210.9 | 0 | 26.6 | 37.5 | 60.8 | 117.1 | 262.6 |
| **128,000** | 4.7 | 18.5 | 36 | 62 | 107.9 | 224.9 | 3.1 | 35.7 | 39 | 65.4 | 126.6 | 252.8 |
| **25,6000** | 9.5 | 23.4 | 40.6 | 72.3 | 118.9 | 233 | 7.9 | 36 | 46.8 | 68.8 | 129.6 | 245.5 |
| **512,000** | 19.9 | 35 | 50 | 82.4 | 136.2 | 231.2 | 22 | 48.3 | 56.2 | 73.3 | 131.2 | 245.1 |
| **1,024,000** | 43.9 | 56.1 | 70.1 | 103 | 155.9 | 264.2 | 42.3 | 77.9 | 66.9 | 86 | 140.4 | 262.4 |
| **2,048,000** | 81.7 | 98.5 | 123.5 | 149.6 | 201.4 | 295.1 | 85.6 | 120.4 | 98.4 | 107.5 | 159.2 | 276.1 |
| **4096000** | 168.8 | 185.9 | 201.3 | 237.5 | 287.2 | 423.1 | 168.8 | 215.6 | 163.6 | 162.5 | 210.9 | 318.8 |
| **8192000** | 351.7 | 357.7 | 393.8 | 427.9 | 476.6 | 614 | 336.1 | 395.8 | 295.2 | 265.6 | 314.1 | 415.5 |
| **16384000** | 693.5 | 696.8 | 736.2 | 765.3 | 815.7 | 1000.2 | 688.9 | 767.4 | 535.7 | 493.8 | 512.4 | 628.1 |
| **32768000** | 1366.8 | 1389.1 | 1398.4 | 1465.8 | 1531.2 | 1660.9 | 1361 | 1429.6 | 1006.2 | 867.3 | 909.3 | 1017.3 |
| **65,536,000** | 2673.3 | 2901.4 | 2785.8 | 2777.8 | 3015.4 | 3070.7 | 2739 | 2815.1 | 1954.7 | 1657.6 | 1715.6 | 1800 |

|  |  |
| --- | --- |
| Ubuntu VM  (1 Processor) | Ubuntu VM  (2 Processors) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Array size  izeize | NT | 1T | 2T | 4T | 8T | 16T | NT | 1T | 2T | 4T | 8T | 16T |
| **1,000** | .02 | 8.6 | 10.1 | 19.3 | 39.8 | 71.6 | .1 | 23 | 26.8 | 58.9 | 106.9 | 220.2 |
| **2,000** | 1.9 | 9.3 | 10.2 | 17 | 41.1 | 71.3 | 1.3 | 17.8 | 22.8 | 52.8 | 107.9 | 225.6 |
| **4,000** | 0.1 | 8.5 | 8.6 | 17.7 | 76.8 | 89.9 | 0.1 | 17.9 | 25.8 | 49.5 | 119.8 | 232 |
| **8,000** | 0.2 | 8 | 12.9 | 17.9 | 42.6 | 87.4 | 0.1 | 23.3 | 25.9 | 72.1 | 124.1 | 225.1 |
| **16,000** | 1.8 | 7.3 | 12.2 | 18.3 | 38.8 | 69.9 | 1.5 | 17.8 | 25.5 | 60 | 118.3 | 206.2 |
| **32,000** | 1.1 | 9.7 | 13 | 20.3 | 53.1 | 114 | 1.3 | 17.5 | 29.9 | 57.3 | 121 | 228.7 |
| **64,000** | 2.8 | 11.1 | 14.3 | 20.6 | 59.5 | 146,8 | 2.9 | 19.3 | 33.3 | 72.5 | 127.2 | 236.9 |
| **128,000** | 7.4 | 19.4 | 17.9 | 23 | 49.4 | 169.1 | 5.1 | 29.9 | 37.2 | 76.3 | 121.1 | 287.1 |
| **25,6000** | 17 | 19.8 | 22.3 | 29.3 | 64.2 | 178.7 | 10.9 | 35.5 | 52.5 | 85.1 | 125.5 | 245.2 |
| **512,000** | 46.9 | 34.9 | 36.6 | 73.9 | 114 | 202.7 | 25.1 | 55.7 | 59.6 | 120 | 154.6 | 267.1 |
| **1,024,000** | 88.5 | 74.1 | 73.3 | 72.7 | 195.2 | 290.9 | 50.9 | 84.5 | 108 | 146.7 | 245.6 | 327.5 |
| **2,048,000** | 169.9 | 153.6 | 136.7 | 239.6 | 285.2 | 389.3 | 112.1 | 159.2 | 185.5 | 212.5 | 280.7 | 391.6 |
| **4096000** | 339.9 | 418.9 | 397.6 | 457.5 | 498.1 | 561.3 | 235.1 | 264.4 | 278.2 | 325.4 | 403.7 | 531.7 |
| **8192000** | 817.1 | 793.9 | 847.3 | 851.8 | 894.2 | 561.3 | 390.8 | 478.6 | 447.1 | 476.8 | 571 | 764.1 |
| **16384000** | 1517.2 | 1509.6 | 1544.7 | 1623.4 | 1655.3 | 1708.2 | 747.5 | 915.8 | 727.8 | 736.9 | 875.9 | 1071.1 |
| **32768000** | 2434 | 2945 | 2973.2 | 3107.7 | 2985.4 | 3300.6 | 1479.6 | 1674.4 | 1324.4 | 1325 | 1403.4 | 1544.6 |
| **65,536,000** | 4850.8 | 5356.6 | 5282.8 | 5323.3 | 5225.3 | 5360.9 | 2952.8 | 3255.5 | 2500.2 | 2538.2 | 2575.2 | 2725.4 |

**Comparing Non-Threaded and Threaded Merge Sorts:**

Random Data Set Averages

**Comparing Windows 10 Virtual Machine to Windows 10 Host:**

**Random Data Set Averages**

|  |  |
| --- | --- |
| Windows 10 Host (NetBeans) | Windows 10 VM (2 Processors) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Array size  izeize | NT | 1T | 2T | 4T | 8T | 16T | NT | 1T | 2T | 4T | 8T | 16T |
| **1,000** | 0.2 | 15.2 | 20.3 | 30.2 | 54.8 | 108.5 | 0 | 40.3 | 37.6 | 64.1 | 121.9 | 248.2 |
| **2,000** | 0.1 | 10.7 | 19.6 | 32.9 | 56.4 | 111.1 | 0 | 32.6 | 40.4 | 67.1 | 124.7 | 250 |
| **4,000** | 0.5 | 11.9 | 17.9 | 29.9 | 55.3 | 138.2 | 1.5 | 28.2 | 40.4 | 64.2 | 122.2 | 256.5 |
| **8,000** | 0.5 | 11.9 | 18.9 | 29.6 | 58 | 113 | 0 | 31.3 | 37.6 | 67.2 | 132.5 | 248.1 |
| **16,000** | 0.6 | 12.8 | 17.9 | 27.8 | 55.4 | 112.1 | 3 | 27.8 | 39.1 | 63.9 | 120.3 | 251.6 |
| **32,000** | 0.9 | 12.4 | 18 | 29.3 | 54.6 | 121.9 | 1.5 | 26.8 | 35.8 | 61.3 | 125.2 | 254.6 |
| **64,000** | 2.2 | 13.3 | 18.8 | 30.1 | 57.6 | 111.5 | 0 | 26.6 | 37.5 | 60.8 | 117.1 | 262.6 |
| **128,000** | 4.1 | 15.1 | 20 | 31.8 | 58 | 112.7 | 3.1 | 35.7 | 39 | 65.4 | 126.6 | 252.8 |
| **25,6000** | 10 | 25.1 | 24.4 | 31.5 | 58.7 | 113.1 | 7.9 | 36 | 46.8 | 68.8 | 129.6 | 245.5 |
| **512,000** | 17.7 | 44.3 | 34.2 | 33.9 | 63.6 | 118.5 | 22 | 48.3 | 56.2 | 73.3 | 131.2 | 245.1 |
| **1,024,000** | 39.4 | 61.8 | 42.9 | 44.4 | 77.4 | 118.2 | 42.3 | 77.9 | 66.9 | 86 | 140.4 | 262.4 |
| **2,048,000** | 78.4 | 114.7 | 62.3 | 63.7 | 84.6 | 133.6 | 85.6 | 120.4 | 98.4 | 107.5 | 159.2 | 276.1 |
| **4096000** | 155 | 202.2 | 111.4 | 104 | 119.9 | 161.6 | 168.8 | 215.6 | 163.6 | 162.5 | 210.9 | 318.8 |
| **8192000** | 313 | 354.4 | 210.7 | 195.7 | 200.5 | 265.9 | 336.1 | 395.8 | 295.2 | 265.6 | 314.1 | 415.5 |
| **16384000** | 618.9 | 671.9 | 382.1 | 363.2 | 355.1 | 393.6 | 688.9 | 767.4 | 535.7 | 493.8 | 512.4 | 628.1 |
| **32768000** | 1227.8 | 1297.3 | 750.6 | 674.3 | 665.7 | 695.7 | 1361 | 1429.6 | 1006.2 | 867.3 | 909.3 | 1017.3 |
| **65,536,000** | 2481.2 | 2528.2 | 1475.6 | 1286.2 | 1327.4 | 1325.8 | 2739 | 2815.1 | 1954.7 | 1657.6 | 1715.6 | 1800 |

Comparing Windows 10 Laptop Run via Command line vs NetBeans

|  |  |
| --- | --- |
| Windows 10 Host  (Netbeans) | Windows 10 Host  (Command line) |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Array size  izeize | NT | 1T | 2T | 4T | 8T | 16T | NT | 1T | 2T | 4T | 8T | 16T |
| **1,000** | 0.2 | 15.2 | 20.3 | 30.2 | 54.8 | 108.5 | 0 | 15.3 | 26 | 32.9 | 65.2 | 121.7 |
| **2,000** | 0.1 | 10.7 | 19.6 | 32.9 | 56.4 | 111.1 | 0 | 9.7 | 20.5 | 33.6 | 71.2 | 120 |
| **4,000** | 0.5 | 11.9 | 17.9 | 29.9 | 55.3 | 138.2 | 0 | 12.7 | 19.3 | 34.2 | 63.4 | 153.9 |
| **8,000** | 0.5 | 11.9 | 18.9 | 29.6 | 58 | 113 | 1.5 | 14.8 | 19.8 | 33.6 | 67.7 | 117 |
| **16,000** | 0.6 | 12.8 | 17.9 | 27.8 | 55.4 | 112.1 | 1.6 | 15.9 | 20.8 | 32.6 | 58.7 | 127.3 |
| **32,000** | 0.9 | 12.4 | 18 | 29.3 | 54.6 | 121.9 | 1.5 | 13.2 | 19.5 | 33.3 | 61.5 | 141.6 |
| **64,000** | 2.2 | 13.3 | 18.8 | 30.1 | 57.6 | 111.5 | 0.3 | 17.5 | 20.6 | 33.6 | 67.8 | 117.4 |
| **128,000** | 4.1 | 15.1 | 20 | 31.8 | 58 | 112.7 | 1.8 | 20.4 | 21.1 | 31.5 | 64.1 | 117.7 |
| **25,6000** | 10 | 25.1 | 24.4 | 31.5 | 58.7 | 113.1 | 13 | 27.5 | 25 | 32.9 | 64.9 | 125.7 |
| **512,000** | 17.7 | 44.3 | 34.2 | 33.9 | 63.6 | 118.5 | 19.5 | 44.4 | 29.2 | 38.5 | 70.5 | 128.1 |
| **1,024,000** | 39.4 | 61.8 | 42.9 | 44.4 | 77.4 | 118.2 | 37.5 | 62.4 | 43.6 | 51.5 | 87.8 | 130.8 |
| **2,048,000** | 78.4 | 114.7 | 62.3 | 63.7 | 84.6 | 133.6 | 76 | 111.1 | **61.9** | 65.6 | 94.1 | 153.2 |
| **4096000** | 155 | 202.2 | 111.4 | 104 | 119.9 | 161.6 | 148.9 | 186.3 | 113.3 | 100.1 | 128.2 | 175.5 |
| **8192000** | 313 | 354.4 | 210.7 | 195.7 | 200.5 | 265.9 | 305.6 | 339.3 | 197.3 | 176.6 | 222.6 | 263 |
| **16384000** | 618.9 | 671.9 | 382.1 | 363.2 | 355.1 | 393.6 | 609.1 | 628 | 384 | 341.3 | 392.6 | 404.8 |
| **32768000** | 1227.8 | 1297.3 | 750.6 | 674.3 | 665.7 | 695.7 | 1219.4 | 1230.6 | 740.9 | 680.8 | 698.7 | 723.2 |
| **65,536,000** | 2481.2 | 2528.2 | 1475.6 | 1286.2 | 1327.4 | 1325.8 | 2453 | 2446 | 1492.3 | 1390.5 | 1324.3 | 1396.5 |

**Comparing Different Data Sets:**

1. Non threaded vs Threaded merge sort performance

Chart of windows random set

Nonthreaded and each thread

Chart of linux random set

Nonthreaded and each thread

1. Windows vs Linux
2. Any difference in different data sets?
3. Data for windows laptop (different number of processors)?

Windows non threaded?

Multi threaded very large data sets?

Strange windows time cost at the beginning of threaded runs (running in loop)

Test windows threaded 1 run at a time to see time cost each run?

Latest Windows 10 and Ubuntu

Netbeans error message importing a project from a Zip file, but project is imported when you go to “open existing project”3

# Future Enhancements

Performance optimizations

External data set

Report generation

Multiple processors

# References

Heineman, George T., et al. *Algorithms in a Nutshell*, O’Reilly Media, Sebastopol, CA, 2016.

Wang, Deming, et al*. An Implementation of Sorting Algorithm Based on Java Multithread*

*Technology*, vol. 1, IEEE, 2012.doi:10.1109/ICCSEE.2012.152.

Concurrency Tutorial by Oracle,

<http://docs.oracle.com/javase/tutorial/essential/concurrency/index.html>, Accessed in September 2016.

Java Documentation by Oracle,

<https://docs.oracle.com/javase/8/docs/api/java/lang/Thread.html>, Accessed in September 2016.

Java Documentation by Oracle,

<https://docs.oracle.com/javase/8/docs/api/java/util/Arrays.html>, Accessed in September 2016.

merge sort code is based on code from: https://courses.cs.washington.edu/courses/cse373/13wi/lectures/03-13/ArraySum.java

stdRandom is based on code from: http://introcs.cs.princeton.edu/java/stdlib/javadoc/StdRandom.html

<http://www.vmware.com/solutions/virtualization.html>

<http://www.infoworld.com/article/2615128/virtualization/review--vmware-workstation-9-vs--virtualbox-4-2.html>

# Appendices

**APPENDIX A**

**Status Reports**

**A.1** Capstone Project Week 5 Status Report for “Comparative Analysis of Multi-Threaded and Non-Threaded Implementations of the Merge Sort Algorithm”

**Date:** 25-September-2016

**Accomplishments**

Activity 1: Hardware and software Installation and testing (22-August-2009 - 25-September-2016)

1. Hardware Procurement – completed for existing hardware. On-going for Virtual Machine requirement.
2. Software Installation – completed for existing hardware. On-going for Virtual Machine requirement.
3. VM installation and testing – completed on existing hardware.

**Current Activities (26-Sept-2016 – 03-Oct-2016)**

Activity 2: Virtual Machine hardware and software procurement and testing and Use Cases

1. Hardware requirements research for Virtual Machines, possible additional hardware procurement and testing.
2. Use Cases

**Challenges**

The challenges are:

1. Additional hardware requirements for Virtual Machines
2. How to obtain or produce a valid data set

The challenges listed above can be resolved and do not place the project at risk.

**Work to be Completed by Oct. 9, 2016**

1. Hardware for Virtual Machine because it is a Capstone Requirement
2. Use Cases
3. Data Set research

**A.2** Capstone Project Week 7 Status Report for “Comparative Analysis of Multi-Threaded and Non-Threaded Implementations of the Merge Sort Algorithm”

**Date:** 03-October-2016

**Accomplishments**

Activity 1: Hardware and software Installation and testing (25-September-2009 - 03-October-2016)

1. Hardware Procurement – completed for existing hardware and for Virtual Machine requirement.
2. Software Installation – completed for existing hardware and for Virtual Machine requirement.
3. VM installation and testing – completed on existing hardware.

 Activity 2: Data Set Research

1. Data Set Research – on-going. Downloaded code from *”Algorithms in a Nutshell”* and researching datasets through the FIT Library.
2. One research paper used several different datasets.

* Random
* Function to return Integers in a range
* Bernoulli function to return 0 or 1
* Geometric function
* Pascal function
* Binomial function
* Zero function
* Unbalanced function

**Current Activities (03-Oct-2016 – 09-Oct-2016)**

Activity 2: Data Set Research and Use Cases

1. Data Set Research and testing code for MergeSort from “*Algorithms in a Nutshell”*
2. Use Cases

**Challenges**

The challenges are:

1. How to obtain or produce a valid data set

The challenges listed above can be resolved and do not place the project at risk.

**Work to be Completed by Oct. 9, 2016**

1. Data Set research
2. Use Cases

**A.3 Capstone Project Week 9 Status Report for** **“Comparative Analysis of Multi-Threaded and Non-Threaded Implementations** **of the Merge Sort Algorithm”**

**Date:** 23-October-2016

**Accomplishments**

Activity 1: Hardware Procurement – complete

Activity 2: Software Installation – complete

Activity 3: VM installation and testing – complete

Activity 4: Uses Cases – in progress

Activity 5: Large Data Set creation – in progress

Activity 6: Non-threaded Merge Sort – in progress

Activity 7: Multi-threaded Merge sort – in progress

Activity 8: Performance Testing – in progress

Activity 9: Data Analysis – in progress

Activity 10: Write report – in progress

**Current Activities (23-Oct-2016 – 06-Nov-2016)**

Activity 4: Uses Cases – in progress

* I have a model for a Use case on paper. My laptop with Visio is at another location. Plan to complete on software this week.

Activity 5: Large Data Set creation – in progress

* Testing existing code for large data set creation
* Coding to create additional data sets based on research results

Activity 6: Non-threaded Merge Sort – in progress

* Testing some existing code
* Researching other merge sort code

Activity 7: Multi-threaded Merge sort – in progress

* Testing some existing code
* Researching other merge sort code

Activity 8: Performance Testing – in progress

Activity 9: Data Analysis – in progress

Activity 10: Write report – in progress

**Challenges**

The challenges are:

1. I am currently testing on my Windows OS on my laptop. I need to Finalizing the code for the data set creation and merge sort algorithm then move to virtual machines and test on the Windows VM and Linux VM.

The challenges listed above can be resolved and do not place the project at risk.

**Work to be Completed by Nov. 6, 2016**

Activity 4: Use Cases - in progress

Activity 5: Large Data Set creation – in progress

Activity 6: Non-threaded Merge Sort – in progress

Activity 7: Multi-threaded Merge sort – in progress

Activity 8: Performance Testing – in progress

Activity 9: Data Analysis – in progress

Activity 10: Write report – in progress

A4. Capstone Project Week 11 Status Report for “Comparative Analysis of Multi-Threaded and Non-Threaded Implementations of the Merge Sort Algorithm”

**Date:** 05-November-2016

**Accomplishments**

Activity 5: Large Data Set creation – in progress

Data Sets I am currently using for testing the performance of my project are:

* Random ( from [https://docs.oracle.com/javase/8/docs/api/java/util/Random.html (Links to an external site.)](https://docs.oracle.com/javase/8/docs/api/java/util/Random.html) )
* StdRandom.java ( from [http://introcs.cs.princeton.edu/java/stdlib (Links to an external site.)](http://introcs.cs.princeton.edu/java/stdlib) )
  + Bernoulli,
  + uniform,
  + Gaussian,
  + Discrete
* Not Tested yet:
  + Zero function (all zeros)

Data Set Size

* Testing sizes of data sets to run on my hardware.
* Have not tested external storage since this will need to run on Virtual Machines

Activity 6: Non-threaded Merge Sort – in progress

* Testing some existing code
* Researching other merge sort code

Activity 7: Multi-threaded Merge sort – in progress

* Testing some existing code
* Researching other merge sort code

Activity 8: Performance Testing – in progress

Activity 9: Data Analysis – in progress

Activity 10: Write report – in progress

Activity 11: Configuration Management

* I have configured GitHub on my laptop to sync with an external site

**Current Activities (05-Nov-2016 – 20-Nov-2016)**

Activity 4: Uses Cases – in progress

Activity 5: Large Data Set creation – in progress

Activity 6: Non-threaded Merge Sort – in progress

Activity 7: Multi-threaded Merge sort – in progress

* Need to test different number of threads

Activity 8: Performance Testing – in progress

Activity 9: Data Analysis – in progress

Activity 10: Write report – in progress

Activity 12: Video  - not tested or completed.

**Challenges**

The challenges are:

1. Added configuration management to the plan for security purposes.

The challenges listed above can be resolved and do not place the project at risk.

**Work to be Completed by Nov. 20, 2016**

Activity 5: Large Data Set creation – in progress

Activity 6: Non-threaded Merge Sort – in progress

Activity 7: Multi-threaded Merge sort – in progress

Activity 8: Performance Testing – in progress

Activity 9: Data Analysis – in progress

Activity 10: Write report – in progress

Activity 12: Video

**APPENDIX B**

**Software Configuration**

**B.1 Steps to install BTMS on a Windows 10 Oracle VirtualBox host**

\*\* Note: BTMS is currently available at <https://drive.google.com/open?id=0B98YurnDqxH0S19Db1c4OHMxZXM>

for this project demonstration.

For manual installation, please follow the instructions below:

Dependency Files:

JDK 6 or higher (<http://www.java.com>)

Apache Tomcat 6 or higher (<http://tomcat.apache.org>)

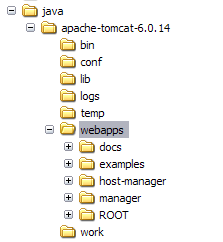
Project File:

DAWS.war (<http://50.56.72.178/Downloads>)

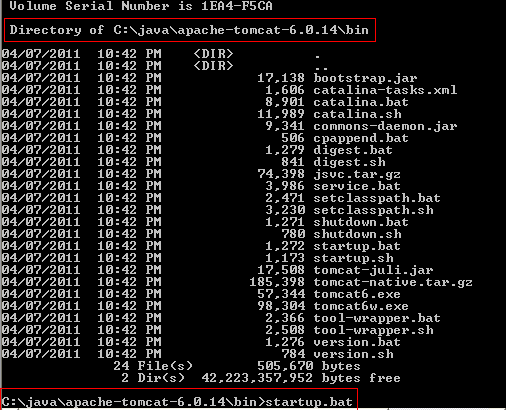
DAWS Server is dependent on the server running JDK 6 or higher. The JRE is not enough, it must be JDK. The DAWS.war file is a java web archive compliant with J2EE standards of deploying web applications. It can be installed on any Java Web Server capable of deploying WAR files. For this demo, Apache Tomcat 6 or higher should be used. You can download Apache Tomcat at <http://tomcat.apache.org/>. Once downloaded, follow the documentation on the Apache Tomcat website on how to install. Be sure to you have Java SDK (not JRE) 6 or higher installed first. If you are unable to receive the DAWS.war file as an attachment to email due to size limitations, go to <http://50.56.72.178/Downloads/> and click on the DAWS.war file to download the file.

Once all resource files are downloaded and JDK 6 and Apache Tomcat are installed, follow these steps to install DAWS:

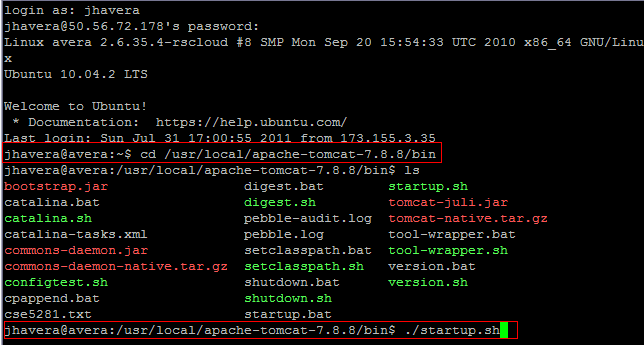
1. Copy DAWS.war to the webapps directory where Tomcat is installed (see image below)



1. Open a command prompt and go to the root directory of where Tomcat is installed and type the following command:
   1. Windows: {$tomcat-home}\bin\startup.bat



* 1. Unix: {$tomcat-home}/bin.startup.sh



1. This will startup Tomcat which will automatically deploy DAWS web application / web service.

Create new jar, put on VMs, create new VM ovas and put on google drive.

========================

BUILD OUTPUT DESCRIPTION

========================

When you build an Java application project that has a main class, the IDE

automatically copies all of the JAR

files on the projects classpath to your projects dist/lib folder. The IDE

also adds each of the JAR files to the Class-Path element in the application

JAR files manifest file (MANIFEST.MF).

To run the project from the command line, go to the dist folder and

type the following:

java -jar "BTMS.jar"

To distribute this project, zip up the dist folder (including the lib folder)

and distribute the ZIP file.

Notes:

\* If two JAR files on the project classpath have the same name, only the first

JAR file is copied to the lib folder.

\* Only JAR files are copied to the lib folder.

If the classpath contains other types of files or folders, these files (folders)

are not copied.

\* If a library on the projects classpath also has a Class-Path element

specified in the manifest,the content of the Class-Path element has to be on

the projects runtime path.

\* To set a main class in a standard Java project, right-click the project node

in the Projects window and choose Properties. Then click Run and enter the

class name in the Main Class field. Alternatively, you can manually type the

class name in the manifest Main-Class element.

**B.2 Steps to install BTMS on Windows 10 Virtual Machine**

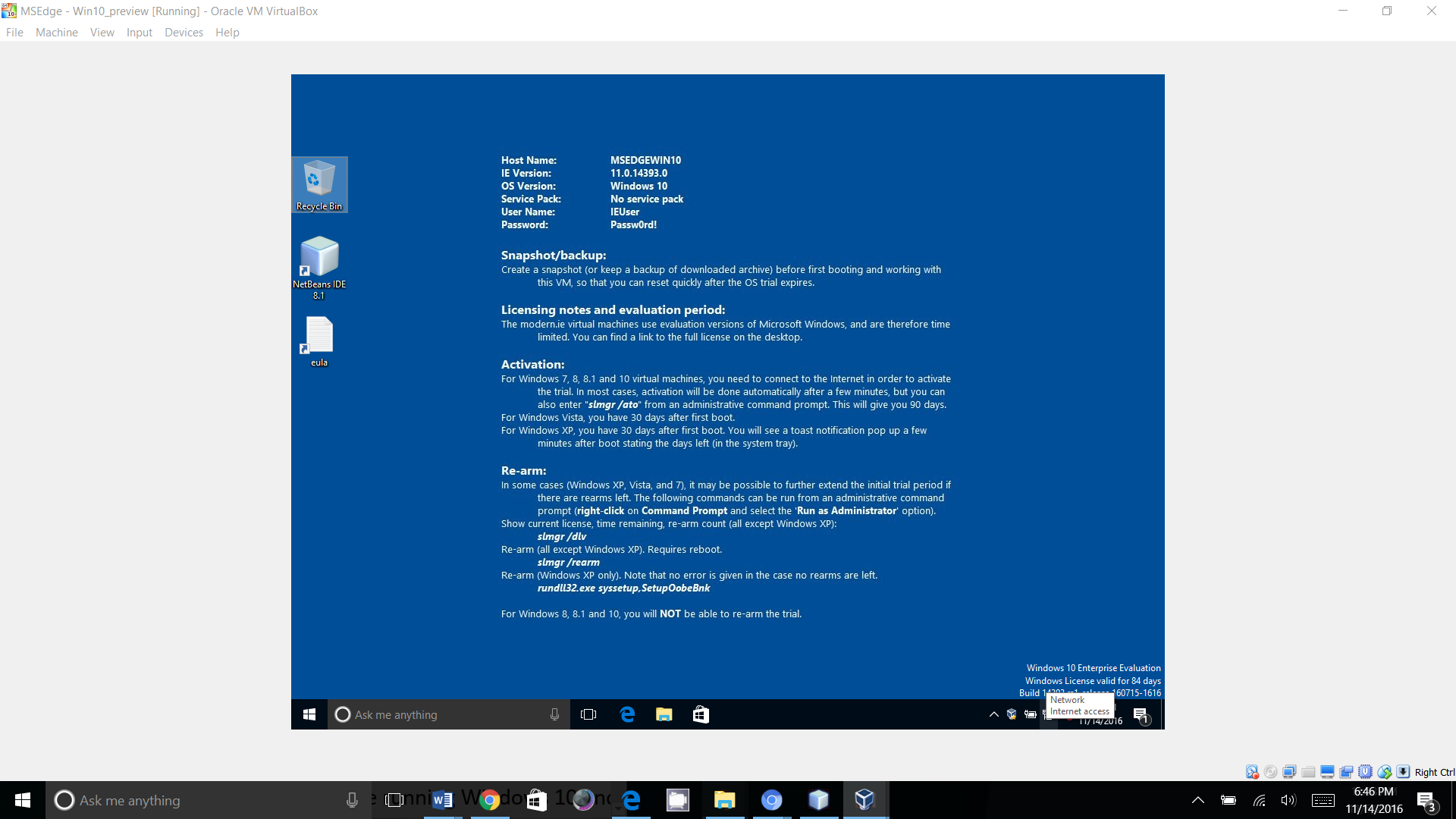
\*\* Note: DAWS Database Server is currently running at jdbc:hsqldb:hsql://50.56.72.178/ for this project demonstration. This is the JDBC connection URL for DAWS Database. The DAWS Server is already configured to connect to the DAWS Database for the demonstration. For manual installation, please follow the instructions below:

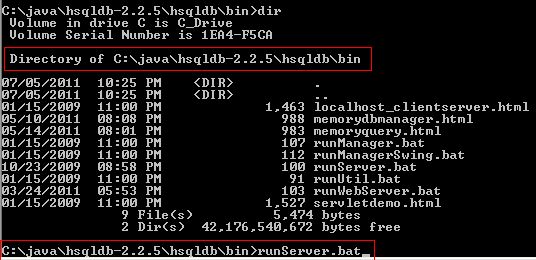
Dependency Files:

JDK 6 or higher ([www.java.com](http://www.java.com))

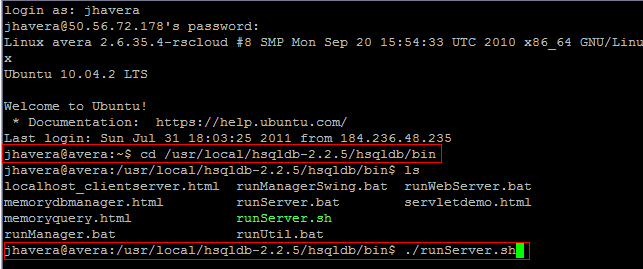
HyperSQL 2.2.5 ([www.hsqldb.org](http://www.hsqldb.org)) or higher (<http://sourceforge.net/projects/hsqldb/files/>)

1. Follow the documentation on the HyperSQL site on how to install and run in server mode.
2. Open a command prompt and go to the root directory of where Tomcat is installed and type the following command:
   1. Windows: {$hsqldb-home}\bin\runServer.bat





* 1. Unix: {$hsqldb-home}/bin.startup.sh



1. Execute the following script to build the TESTDATA table:

CREATE TABLE TESTDATA (

ID INTEGER,

FIRSTNAME VARCHAR(50),

LASTNAME VARCHAR(50),

STREET VARCHAR(50),

CITY VARCHAR(50)

);

1. Data can be loaded into the table using the DAWS RCP client application.

**B.3 Steps to install BTMS on Ubuntu Virtual Machine**

**ubuntuVMuser**

**Passw0rd!**

Dependency Files:

JDK 6 or higher (<http://www.java.com>)

Project File:

DAWS-RCP.zip (<http://50.56.72.178/Downloads>)

DAWS-RCP is a client application for Windows designed to demonstrate a practical purpose and capability of DAWS. Although it is a Java application, a Unix version was not built for this demonstration. To install, please follow these steps:

1. Unzip DAWS-RPC.zip file to your hard drive on a Windows machine.
2. Go to the install directory and find eclipse.exe in the eclipse subdirectory.
3. Double-click the eclipse.exe, and you should see a brief splash screen followed by the main client application:

**APPENDIX C**

**User’s Manual**

BTMS is a Java application to test non-threaded and threaded Merge Sort Performance. Once you have it up and running, follow these steps to demonstrate the capabilities:

1. Select TESTDATA in the Data Sets Available combo box:
2. Click Get Data button:
3. See results of how long the data file was produced and transferred to the client using DAWS. The screenshot below are the results of running DAWS over a cellular network driving down the road using TESTDATA with 10,000 rows. The results will vary based on bandwidth. For most broadband connection, the performance will be less than 2 seconds.
4. Optional step is to click the Display Data to see the data once it is parsed on the client side. A time is displayed to perform this function. This would be similar if the client wanted to parse the data file and persist the data to its own database:

**Experiment 1: Windows 10 VM 1 Processor**

**Random Data Set**

**No threads**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2000 | 16 | 0 | 0 | 0 | 16 | 0 | 0 | 0 | 0 | 0 | 3.2 | 6.4 |
| 4000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 32000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 64000 | 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1.5 | 4.5 |
| 128000 | 0 | 0 | 0 | 15 | 16 | 0 | 0 | 0 | 0 | 16 | 4.7 | 7.184010022 |
| 256000 | 16 | 0 | 16 | 0 | 0 | 16 | 16 | 15 | 16 | 0 | 9.5 | 7.762087348 |
| 512000 | 15 | 16 | 15 | 32 | 30 | 15 | 15 | 15 | 15 | 31 | 19.9 | 7.286288493 |
| 1024000 | 47 | 47 | 47 | 46 | 47 | 32 | 47 | 47 | 47 | 32 | 43.9 | 5.957348403 |
| 2048000 | 93 | 78 | 78 | 78 | 95 | 95 | 79 | 79 | 63 | 79 | 81.7 | 9.45568612 |
| 4096000 | 172 | 156 | 156 | 218 | 172 | 172 | 171 | 156 | 157 | 158 | 168.8 | 17.8986033 |
| 8192000 | 312 | 375 | 360 | 406 | 344 | 359 | 328 | 328 | 376 | 329 | 351.7 | 27.3461149 |
| 16384000 | 625 | 718 | 703 | 703 | 688 | 718 | 671 | 672 | 734 | 703 | 693.5 | 29.6690074 |
| 32768000 | 1234 | 1468 | 1391 | 1468 | 1438 | 1421 | 1328 | 1312 | 1312 | 1296 | 1366.8 | 76.96466722 |
| 65536000 | 2532 | 2703 | 2704 | 2937 | 2749 | 2610 | 2719 | 2624 | 2531 | 2624 | 2673.3 | 113.2537417 |
| 1 thread |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 31 | 0 | 16 | 16 | 15 | 0 | 16 | 16 | 16 | 15 | 14.1 | 8.383913167 |
| 2000 | 16 | 16 | 16 | 16 | 16 | 16 | 15 | 0 | 15 | 16 | 14.2 | 4.749736835 |
| 4000 | 31 | 15 | 15 | 15 | 16 | 16 | 18 | 15 | 16 | 16 | 17.3 | 4.64865572 |
| 8000 | 16 | 16 | 0 | 0 | 15 | 15 | 13 | 16 | 0 | 15 | 10.6 | 6.988562084 |
| 16000 | 15 | 16 | 0 | 16 | 16 | 16 | 16 | 16 | 16 | 16 | 14.3 | 4.775981575 |
| 32000 | 16 | 16 | 15 | 16 | 15 | 16 | 16 | 16 | 15 | 0 | 14.1 | 4.721228654 |
| 64000 | 16 | 15 | 16 | 15 | 16 | 15 | 15 | 15 | 16 | 16 | 15.5 | 0.5 |
| 128000 | 15 | 15 | 16 | 16 | 15 | 31 | 16 | 16 | 15 | 30 | 18.5 | 6.020797289 |
| 256000 | 31 | 32 | 14 | 31 | 32 | 16 | 15 | 15 | 32 | 16 | 23.4 | 8.224354078 |
| 512000 | 32 | 32 | 47 | 34 | 31 | 47 | 32 | 32 | 32 | 31 | 35 | 6.049793385 |
| 1024000 | 47 | 63 | 62 | 47 | 63 | 62 | 62 | 47 | 61 | 47 | 56.1 | 7.449161027 |
| 2048000 | 95 | 109 | 94 | 110 | 93 | 94 | 94 | 93 | 94 | 109 | 98.5 | 7.116881339 |
| 4096000 | 187 | 203 | 188 | 203 | 188 | 188 | 187 | 171 | 172 | 172 | 185.9 | 10.99499886 |
| 8192000 | 329 | 360 | 375 | 421 | 359 | 359 | 327 | 360 | 328 | 359 | 357.7 | 26.40094695 |
| 16384000 | 640 | 671 | 703 | 797 | 703 | 813 | 657 | 688 | 640 | 656 | 696.8 | 58.36403002 |
| 32768000 | 1250 | 1345 | 1329 | 1672 | 1485 | 1453 | 1312 | 1359 | 1328 | 1358 | 1389.1 | 113.9161534 |
| 65536000 | 4531 | 2782 | 2625 | 2766 | 2781 | 2780 | 2750 | 2671 | 2688 | 2640 | 2901.4 | 546.2318555 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 30 | 31 | 15 | 31 | 31 | 32 | 16 | 32 | 16 | 31 | 26.5 | 7.116881339 |
| 2000 | 31 | 16 | 31 | 16 | 16 | 31 | 31 | 16 | 31 | 16 | 23.5 | 7.5 |
| 4000 | 31 | 31 | 32 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 31.1 | 0.3 |
| 8000 | 16 | 31 | 15 | 32 | 31 | 32 | 19 | 31 | 16 | 16 | 23.9 | 7.569015788 |
| 16000 | 32 | 16 | 31 | 31 | 31 | 15 | 16 | 16 | 31 | 32 | 25.1 | 7.647875522 |
| 32000 | 30 | 31 | 16 | 31 | 16 | 31 | 31 | 16 | 31 | 30 | 26.3 | 6.753517602 |
| 64000 | 47 | 31 | 31 | 47 | 32 | 32 | 32 | 31 | 32 | 32 | 34.7 | 6.165225057 |
| 128000 | 32 | 32 | 47 | 31 | 47 | 47 | 31 | 31 | 31 | 31 | 36 | 7.211102551 |
| 256000 | 46 | 47 | 31 | 47 | 31 | 46 | 46 | 48 | 32 | 32 | 40.6 | 7.45922248 |
| 512000 | 48 | 46 | 47 | 62 | 47 | 48 | 48 | 46 | 46 | 62 | 50 | 6.049793385 |
| 1024000 | 77 | 78 | 62 | 79 | 78 | 62 | 62 | 78 | 63 | 62 | 70.1 | 7.917701687 |
| 2048000 | 110 | 125 | 109 | 188 | 126 | 125 | 110 | 108 | 125 | 109 | 123.5 | 22.8002193 |
| 4096000 | 187 | 202 | 188 | 234 | 203 | 219 | 203 | 187 | 203 | 187 | 201.3 | 14.77193285 |
| 8192000 | 438 | 360 | 376 | 421 | 406 | 406 | 374 | 406 | 375 | 376 | 393.8 | 23.79411692 |
| 16384000 | 751 | 719 | 719 | 812 | 813 | 750 | 688 | 703 | 703 | 704 | 736.2 | 42.67270791 |
| 32768000 | 1392 | 1375 | 1390 | 1562 | 1422 | 1406 | 1375 | 1329 | 1375 | 1358 | 1398.4 | 59.6509849 |
| 65536000 | 2859 | 2827 | 2796 | 3000 | 2844 | 2766 | 2720 | 2750 | 2625 | 2671 | 2785.8 | 100.7311273 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 threads |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 63 | 48 | 62 | 47 | 47 | 47 | 47 | 47 | 46 | 47 | 50.1 | 6.220128616 |
| 2000 | 77 | 47 | 47 | 63 | 62 | 63 | 48 | 48 | 48 | 47 | 55 | 10 |
| 4000 | 48 | 62 | 47 | 46 | 47 | 46 | 63 | 46 | 62 | 62 | 52.9 | 7.66093989 |
| 8000 | 62 | 47 | 46 | 63 | 63 | 63 | 46 | 63 | 47 | 48 | 54.8 | 8.022468448 |
| 16000 | 62 | 47 | 48 | 48 | 46 | 62 | 62 | 47 | 47 | 62 | 53.1 | 7.286288493 |
| 32000 | 47 | 47 | 62 | 62 | 63 | 79 | 48 | 62 | 62 | 47 | 57.9 | 9.964436763 |
| 64000 | 63 | 62 | 47 | 46 | 47 | 47 | 64 | 47 | 48 | 62 | 53.3 | 7.746612163 |
| 128000 | 46 | 63 | 62 | 62 | 62 | 62 | 60 | 78 | 62 | 63 | 62 | 7.197221686 |
| 256000 | 79 | 63 | 63 | 94 | 63 | 63 | 63 | 109 | 63 | 63 | 72.3 | 15.71018778 |
| 512000 | 62 | 77 | 78 | 109 | 77 | 96 | 77 | 94 | 77 | 77 | 82.4 | 12.68227109 |
| 1024000 | 93 | 94 | 93 | 125 | 110 | 107 | 95 | 125 | 94 | 94 | 103 | 12.40967365 |
| 2048000 | 156 | 140 | 140 | 171 | 140 | 156 | 140 | 172 | 140 | 141 | 149.6 | 12.55547689 |
| 4096000 | 219 | 235 | 216 | 297 | 235 | 250 | 235 | 234 | 219 | 235 | 237.5 | 22.0918537 |
| 8192000 | 516 | 406 | 406 | 437 | 453 | 421 | 406 | 406 | 391 | 437 | 427.9 | 34.47448332 |
| 16384000 | 734 | 781 | 765 | 906 | 781 | 749 | 734 | 719 | 734 | 750 | 765.3 | 50.86462425 |
| 32768000 | 1703 | 1438 | 1454 | 1579 | 1501 | 1406 | 1406 | 1359 | 1390 | 1422 | 1465.8 | 98.71757696 |
| 65536000 | 2953 | 2719 | 2781 | 2890 | 2859 | 2795 | 2703 | 2703 | 2641 | 2734 | 2777.8 | 92.44652508 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 94 | 109 | 109 | 109 | 109 | 109 | 110 | 94 | 94 | 109 | 104.6 | 6.945502142 |
| 2000 | 94 | 94 | 94 | 110 | 109 | 109 | 94 | 93 | 94 | 95 | 98.6 | 7.045565982 |
| 4000 | 125 | 110 | 109 | 140 | 111 | 110 | 109 | 110 | 109 | 93 | 112.6 | 11.62067124 |
| 8000 | 110 | 94 | 94 | 140 | 93 | 110 | 109 | 94 | 94 | 109 | 104.7 | 13.90719238 |
| 16000 | 94 | 109 | 109 | 125 | 109 | 109 | 93 | 109 | 94 | 95 | 104.6 | 9.820386958 |
| 32000 | 109 | 110 | 110 | 141 | 110 | 93 | 110 | 110 | 109 | 124 | 112.6 | 11.74904251 |
| 64000 | 109 | 109 | 172 | 125 | 109 | 109 | 109 | 93 | 94 | 109 | 113.8 | 21.15561391 |
| 128000 | 109 | 93 | 109 | 126 | 125 | 110 | 110 | 109 | 93 | 95 | 107.9 | 11.14854251 |
| 256000 | 125 | 109 | 128 | 124 | 109 | 125 | 109 | 126 | 125 | 109 | 118.9 | 8.141867108 |
| 512000 | 126 | 126 | 138 | 157 | 188 | 127 | 125 | 124 | 141 | 110 | 136.2 | 20.96568625 |
| 1024000 | 202 | 138 | 140 | 219 | 156 | 141 | 141 | 141 | 141 | 140 | 155.9 | 27.96587206 |
| 2048000 | 188 | 187 | 234 | 266 | 203 | 187 | 186 | 188 | 188 | 187 | 201.4 | 25.76121115 |
| 4096000 | 282 | 328 | 281 | 329 | 297 | 296 | 265 | 263 | 266 | 265 | 287.2 | 23.81512125 |
| 8192000 | 453 | 500 | 469 | 578 | 484 | 469 | 437 | 469 | 453 | 454 | 476.6 | 37.74970199 |
| 16384000 | 828 | 828 | 813 | 859 | 860 | 798 | 812 | 781 | 797 | 781 | 815.7 | 26.89256403 |
| 32768000 | 1562 | 1469 | 1688 | 1781 | 1531 | 1531 | 1453 | 1421 | 1453 | 1423 | 1531.2 | 113.0865156 |
| 65536000 | 4094 | 3078 | 3031 | 3063 | 3031 | 2812 | 2828 | 2719 | 2796 | 2702 | 3015.4 | 384.8049376 |
| 16 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 204 | 235 | 189 | 251 | 203 | 266 | 203 | 204 | 203 | 187 | 214.5 | 25.35448678 |
| 2000 | 203 | 202 | 218 | 234 | 219 | 218 | 203 | 234 | 203 | 204 | 213.8 | 12.09793371 |
| 4000 | 203 | 221 | 203 | 219 | 250 | 328 | 187 | 204 | 188 | 187 | 219 | 40.73327878 |
| 8000 | 218 | 201 | 219 | 266 | 219 | 235 | 204 | 203 | 218 | 203 | 218.6 | 18.77871135 |
| 16000 | 250 | 203 | 234 | 265 | 219 | 281 | 187 | 203 | 187 | 188 | 221.7 | 32.54857908 |
| 32000 | 251 | 188 | 203 | 234 | 203 | 282 | 203 | 203 | 203 | 203 | 217.3 | 27.72381648 |
| 64000 | 218 | 203 | 203 | 204 | 218 | 281 | 203 | 188 | 204 | 187 | 210.9 | 25.28418478 |
| 128000 | 219 | 203 | 219 | 249 | 219 | 266 | 203 | 234 | 218 | 219 | 224.9 | 18.70534683 |
| 256000 | 203 | 219 | 219 | 204 | 281 | 391 | 203 | 203 | 204 | 203 | 233 | 57.39512174 |
| 512000 | 219 | 219 | 234 | 218 | 235 | 266 | 234 | 234 | 234 | 219 | 231.2 | 13.70255451 |
| 1024000 | 296 | 281 | 250 | 250 | 266 | 281 | 251 | 249 | 268 | 250 | 264.2 | 16.13567476 |
| 2048000 | 297 | 297 | 313 | 281 | 297 | 311 | 282 | 298 | 294 | 281 | 295.1 | 10.78378412 |
| 4096000 | 374 | 391 | 403 | 390 | 593 | 516 | 375 | 438 | 376 | 375 | 423.1 | 70.35971859 |
| 8192000 | 578 | 563 | 641 | 577 | 751 | 765 | 578 | 578 | 547 | 562 | 614 | 75.72978278 |
| 16384000 | 968 | 938 | 1329 | 1032 | 953 | 1203 | 891 | 891 | 875 | 922 | 1000.2 | 142.4856484 |
| 32768000 | 1640 | 1688 | 1640 | 1859 | 1673 | 1859 | 1546 | 1547 | 1579 | 1578 | 1660.9 | 109.5430965 |
| 65536000 | 2922 | 3001 | 2953 | 3532 | 2969 | 3625 | 2860 | 2922 | 2907 | 3016 | 3070.7 | 258.3617812 |

**Experiment 2: Random Data Set: Ubuntu VM 1 Processor**

**No threads**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0.4 | 0.663324958 |
| 2000 | 19 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 2.1 | 5.647123161 |
| 4000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8000 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0.2 | 0.4 |
| 16000 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0.5 | 0.5 |
| 32000 | 1 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 1 | 1 | 1.2 | 0.4 |
| 64000 | 2 | 2 | 2 | 3 | 2 | 2 | 2 | 2 | 7 | 2 | 2.6 | 1.496662955 |
| 128000 | 32 | 4 | 5 | 4 | 4 | 5 | 4 | 5 | 4 | 4 | 7.1 | 8.312039461 |
| 256000 | 27 | 9 | 9 | 9 | 9 | 9 | 8 | 12 | 9 | 9 | 11 | 5.422176685 |
| 512000 | 164 | 19 | 18 | 18 | 18 | 18 | 17 | 20 | 19 | 19 | 33 | 43.67379077 |
| 1024000 | 410 | 38 | 40 | 42 | 39 | 39 | 37 | 46 | 40 | 38 | 76.9 | 111.0598487 |
| 2048000 | 1085 | 79 | 76 | 126 | 79 | 78 | 85 | 109 | 81 | 79 | 187.7 | 299.5059432 |
| 4096000 | 1935 | 199 | 192 | 162 | 195 | 241 | 206 | 312 | 187 | 187 | 381.6 | 519.2853166 |
| 8192000 | 2641 | 711 | 689 | 673 | 703 | 690 | 695 | 791 | 760 | 670 | 902.3 | 580.6990701 |
| 16384000 | 2438 | 1241 | 1446 | 1258 | 1268 | 1296 | 1427 | 1427 | 1401 | 1326 | 1453 | 336.4968945 |
| 32768000 | 4706 | 2488 | 2889 | 2751 | 2533 | 2619 | 2740 | 3175 | 3129 | 2870 | 2990 | 611.630444 |
| 65536000 | 7418 | 4820 | 5265 | 5127 | 4978 | 5113 | 5180 | 5169 | 5224 | 5169 | 5346 | 701.3695246 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 thread |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 53 | 5 | 4 | 4 | 6 | 5 | 5 | 4 | 5 | 5 | 9.6 | 14.47895024 |
| 2000 | 54 | 5 | 4 | 5 | 4 | 5 | 4 | 5 | 5 | 5 | 9.6 | 14.80675522 |
| 4000 | 39 | 4 | 5 | 5 | 5 | 5 | 4 | 8 | 4 | 5 | 8.4 | 10.25865488 |
| 8000 | 31 | 5 | 5 | 5 | 5 | 5 | 5 | 6 | 5 | 5 | 7.7 | 7.772387021 |
| 16000 | 50 | 5 | 5 | 5 | 5 | 5 | 6 | 5 | 5 | 5 | 9.6 | 13.46996659 |
| 32000 | 40 | 6 | 6 | 6 | 6 | 5 | 6 | 9 | 5 | 6 | 9.5 | 10.22007828 |
| 64000 | 43 | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 6 | 13 | 11.1 | 10.79305332 |
| 128000 | 38 | 9 | 9 | 9 | 8 | 9 | 8 | 13 | 9 | 9 | 12.1 | 8.734414691 |
| 256000 | 143 | 13 | 14 | 14 | 19 | 14 | 13 | 17 | 14 | 14 | 27.5 | 38.54153604 |
| 512000 | 82 | 22 | 23 | 24 | 30 | 24 | 22 | 31 | 22 | 23 | 30.3 | 17.51028269 |
| 1024000 | 302 | 42 | 43 | 84 | 48 | 47 | 43 | 57 | 42 | 45 | 75.3 | 76.53241143 |
| 2048000 | 445 | 110 | 83 | 114 | 107 | 124 | 108 | 117 | 116 | 102 | 142.6 | 101.3412058 |
| 4096000 | 600 | 230 | 181 | 191 | 278 | 187 | 156 | 467 | 329 | 284 | 290.3 | 135.2686586 |
| 8192000 | 1359 | 707 | 885 | 757 | 792 | 735 | 851 | 932 | 821 | 857 | 869.6 | 176.0018182 |
| 16384000 | 2355 | 1452 | 1829 | 1366 | 1497 | 1295 | 1472 | 1604 | 1566 | 1492 | 1593 | 288.3507586 |
| 32768000 | 4794 | 2186 | 2963 | 2633 | 2978 | 2435 | 2670 | 3401 | 2884 | 2984 | 2993 | 679.6832792 |
| 65536000 | 6809 | 5058 | 5323 | 4994 | 5221 | 4851 | 5063 | 5610 | 5376 | 5106 | 5341 | 530.7015074 |
| 2 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 15 | 8 | 8 | 8 | 8 | 12 | 7 | 8 | 8 | 7 | 8.9 | 2.42693222 |
| 2000 | 16 | 9 | 8 | 8 | 7 | 7 | 7 | 8 | 7 | 8 | 8.5 | 2.578759392 |
| 4000 | 16 | 8 | 8 | 20 | 8 | 9 | 8 | 16 | 8 | 7 | 10.8 | 4.422668877 |
| 8000 | 12 | 9 | 8 | 8 | 8 | 10 | 8 | 17 | 7 | 7 | 9.4 | 2.905167809 |
| 16000 | 10 | 10 | 9 | 9 | 11 | 11 | 8 | 18 | 8 | 8 | 10.2 | 2.821347196 |
| 32000 | 8 | 9 | 8 | 9 | 9 | 10 | 9 | 9 | 14 | 8 | 9.3 | 1.676305461 |
| 64000 | 10 | 11 | 10 | 11 | 10 | 11 | 10 | 11 | 10 | 9 | 10.3 | 0.640312424 |
| 128000 | 22 | 13 | 13 | 13 | 12 | 13 | 12 | 14 | 12 | 12 | 13.6 | 2.870540019 |
| 256000 | 37 | 24 | 17 | 17 | 17 | 17 | 16 | 19 | 17 | 16 | 19.7 | 6.181423784 |
| 512000 | 98 | 27 | 28 | 27 | 27 | 27 | 49 | 28 | 26 | 26 | 36.3 | 21.5965275 |
| 1024000 | 130 | 49 | 46 | 93 | 48 | 60 | 48 | 54 | 47 | 47 | 62.2 | 26.30513258 |
| 2048000 | 370 | 110 | 117 | 92 | 114 | 108 | 104 | 124 | 620 | 117 | **187.6** | 163.7548167 |
| 4096000 | 691 | 431 | 282 | 326 | 381 | 304 | 226 | 400 | 1041 | 223 | 430.5 | 240.3752275 |
| 8192000 | 1619 | 787 | 696 | 705 | 840 | 792 | 785 | 961 | 1554 | 728 | 946.7 | 328.1877664 |
| 16384000 | 2461 | 1343 | 1313 | 1306 | 1497 | 1276 | 1348 | 1932 | 2151 | 1378 | 1601 | 402.2897588 |
| 32768000 | 4642 | 2443 | 3102 | 2506 | 2598 | 2620 | 2796 | 3566 | 3010 | 2740 | 3002 | 632.1768819 |
| 65536000 | 6305 | 4866 | 5483 | 4911 | 5012 | 4926 | 5051 | 5762 | 5576 | 5063 | 5296 | 448.7313784 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 23 | 26 | 20 | 14 | 16 | 14 | 15 | 17 | 14 | 15 | 17.4 | 4.004996879 |
| 2000 | 13 | 27 | 20 | 15 | 17 | 14 | 30 | 17 | 15 | 14 | 18.2 | 5.528109984 |
| 4000 | 13 | 15 | 34 | 15 | 17 | 14 | 26 | 17 | 15 | 15 | 18.1 | 6.31585307 |
| 8000 | 23 | 15 | 21 | 14 | 15 | 14 | 20 | 20 | 15 | 15 | 17.2 | 3.218695388 |
| 16000 | 14 | 16 | 20 | 15 | 16 | 14 | 17 | 18 | 15 | 15 | 16 | 1.788854382 |
| 32000 | 15 | 16 | 24 | 15 | 17 | 15 | 16 | 21 | 16 | 15 | 17 | 2.898275349 |
| 64000 | 20 | 17 | 20 | 17 | 18 | 16 | 17 | 25 | 17 | 17 | 18.4 | 2.537715508 |
| 128000 | 24 | 20 | 28 | 20 | 20 | 18 | 19 | 25 | 19 | 19 | 21.2 | 3.12409987 |
| 256000 | 35 | 26 | 32 | 24 | 29 | 23 | 24 | 34 | 25 | 24 | 27.6 | 4.317406629 |
| 512000 | 297 | 59 | 39 | 60 | 69 | 33 | 64 | 82 | 35 | 35 | 77.3 | 74.95205134 |
| 1024000 | 228 | 58 | 98 | 96 | 64 | 107 | 59 | 76 | 87 | 88 | 96.1 | 46.83043882 |
| 2048000 | 370 | 118 | 194 | 103 | 181 | 94 | 117 | 297 | 99 | 96 | 166.9 | 91.11800042 |
| 4096000 | 738 | 381 | 514 | 341 | 471 | 400 | 437 | 550 | 478 | 440 | 475 | 105.5585146 |
| 8192000 | 1375 | 695 | 882 | 827 | 870 | 924 | 833 | 1001 | 740 | 753 | 890 | 183.215174 |
| 16384000 | 2635 | 1301 | 1891 | 1595 | 1591 | 1308 | 1434 | 1960 | 1471 | 1315 | 1650 | 395.0441621 |
| 32768000 | 5017 | 2657 | 3620 | 2830 | 2966 | 2710 | 2959 | 3887 | 2987 | 2656 | 3229 | 711.8635333 |
| 65536000 | 6637 | 5099 | 5429 | 5204 | 5585 | 5013 | 5173 | 6231 | 5333 | 5188 | 5489 | 505.5667711 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 39 | 30 | 27 | 26 | 29 | 27 | 27 | 29 | 27 | 31 | 29.2 | 3.6 |
| 2000 | 54 | 33 | 28 | 27 | 28 | 28 | 26 | 30 | 28 | 36 | 31.8 | 7.93473377 |
| 4000 | 48 | 26 | 28 | 26 | 37 | 28 | 27 | 29 | 26 | 28 | 30.3 | 6.649060084 |
| 8000 | 130 | 26 | 28 | 26 | 31 | 28 | 30 | 49 | 30 | 28 | 40.6 | 30.46703136 |
| 16000 | 106 | 30 | 28 | 26 | 29 | 41 | 28 | 59 | 27 | 28 | 40.2 | 23.94911272 |
| 32000 | 75 | 28 | 28 | 27 | 34 | 42 | 27 | 43 | 28 | 29 | 36.1 | 14.1876707 |
| 64000 | 219 | 42 | 31 | 74 | 39 | 62 | 29 | 42 | 29 | 54 | 62.1 | 54.15246993 |
| 128000 | 147 | 49 | 39 | 39 | 34 | 35 | 60 | 63 | 60 | 51 | 57.7 | 31.48031131 |
| 256000 | 171 | 43 | 40 | 37 | 42 | 39 | 38 | 130 | 45 | 38 | 62.3 | 45.10443437 |
| 512000 | 192 | 54 | 78 | 47 | 73 | 49 | 47 | 190 | 51 | 46 | 82.7 | 55.16892241 |
| 1024000 | 326 | 178 | 95 | 68 | 130 | 101 | 71 | 241 | 128 | 97 | 143.5 | 78.55093889 |
| 2048000 | 475 | 309 | 213 | 236 | 218 | 293 | 256 | 349 | 285 | 287 | 292.1 | 72.97321426 |
| 4096000 | 812 | 535 | 398 | 438 | 454 | 416 | 452 | 569 | 491 | 449 | 501.4 | 114.7677655 |
| 8192000 | 1470 | 748 | 798 | 838 | 706 | 718 | 883 | 970 | 949 | 862 | 894.2 | 210.1688845 |
| 16384000 | 2521 | 1314 | 1659 | 1598 | 1415 | 1459 | 1525 | 1791 | 1322 | 1541 | 1615 | 332.9258927 |
| 32768000 | 4982 | 2497 | 2504 | 2945 | 2751 | 2811 | 3103 | 2866 | 2765 | 2884 | 3011 | 680.0129116 |
| 65536000 | 6634 | 4953 | 4981 | 5220 | 4855 | 4996 | 5295 | 5952 | 5296 | 5075 | 5326 | 526.0923968 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 16 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 154 | 53 | 55 | 50 | 54 | 67 | 52 | 90 | 55 | 53 | 68.3 | 30.73125445 |
| 2000 | 125 | 53 | 54 | 51 | 80 | 57 | 52 | 65 | 92 | 52 | 68.1 | 23.0670761 |
| 4000 | 104 | 54 | 55 | 50 | 57 | 70 | 51 | 54 | 68 | 52 | 61.5 | 15.58364527 |
| 8000 | 195 | 82 | 80 | 63 | 57 | 82 | 74 | 85 | 76 | 69 | 86.3 | 37.21840942 |
| 16000 | 182 | 101 | 60 | 70 | 78 | 54 | 63 | 57 | 56 | 66 | 78.7 | 36.87560169 |
| 32000 | 248 | 55 | 56 | 53 | 59 | 52 | 53 | 56 | 57 | 54 | 74.3 | 57.93453202 |
| 64000 | 333 | 639 | 58 | 54 | 66 | 55 | 68 | 106 | 132 | 57 | 156.8 | 179.8726216 |
| 128000 | 299 | 133 | 134 | 122 | 143 | 134 | 136 | 164 | 139 | 115 | 151.9 | 50.53404793 |
| 256000 | 275 | 155 | 135 | 136 | 174 | 138 | 104 | 178 | 154 | 155 | 160.4 | 43.18610888 |
| 512000 | 338 | 169 | 200 | 202 | 210 | 162 | 174 | 191 | 174 | 203 | 202.3 | 47.89791227 |
| 1024000 | 423 | 278 | 208 | 228 | 202 | 157 | 227 | 212 | 249 | 186 | 237 | 69.49388462 |
| 2048000 | 633 | 358 | 329 | 348 | 300 | 305 | 359 | 442 | 392 | 266 | 373.2 | 98.54217371 |
| 4096000 | 976 | 515 | 509 | 531 | 521 | 499 | 487 | 523 | 546 | 512 | 561.9 | 138.8981281 |
| 8192000 | 1600 | 847 | 1033 | 951 | 906 | 741 | 871 | 986 | 881 | 860 | 967.6 | 224.2579764 |
| 16384000 | 2225 | 1629 | 2243 | 1316 | 1613 | 1368 | 1377 | 1626 | 1648 | 1363 | 1641 | 321.215753 |
| 32768000 | 4933 | 3208 | 3133 | 2606 | 2968 | 2753 | 3008 | 3126 | 3025 | 2682 | 3144 | 626.4553935 |
| 65536000 | 6549 | 5292 | 5275 | 4717 | 5338 | 5165 | 5306 | 5887 | 5139 | 5097 | 5377 | 476.8081899 |

**Experiment 3:Random Data Set: Windows 10 VM 2 Processors: All Runs, ms**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4000 | 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1.5 | 4.5 |
| 8000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16000 | 0 | 0 | 0 | 0 | 0 | 0 | 15 | 0 | 15 | 0 | 3 | 6 |
| 32000 | 0 | 0 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 0 | 1.5 | 4.5 |
| 64000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 128000 | 0 | 0 | 0 | 15 | 0 | 0 | 0 | 16 | 0 | 0 | 3.1 | 6.20403095 |
| 256000 | 16 | 0 | 15 | 0 | 0 | 16 | 16 | 0 | 16 | 0 | 7.9 | 7.90506167 |
| 512000 | 31 | 15 | 32 | 16 | 16 | 16 | 31 | 16 | 31 | 16 | 22 | 7.56306816 |
| 1024000 | 47 | 32 | 46 | 48 | 47 | 47 | 47 | 32 | 47 | 30 | 42.3 | 7.21179589 |
| 2048000 | 141 | 78 | 77 | 78 | 94 | 77 | 78 | 77 | 78 | 78 | 85.6 | 19.10602 |
| 4096000 | 156 | 172 | 171 | 157 | 172 | 172 | 187 | 172 | 157 | 172 | 168.8 | 9.10823803 |
| 8192000 | 329 | 344 | 328 | 344 | 343 | 313 | 344 | 328 | 328 | 360 | 336.1 | 12.5972219 |
| 16384000 | 657 | 656 | 655 | 734 | 703 | 687 | 735 | 702 | 642 | 718 | 688.9 | 32.907294 |
| 32768000 | 1281 | 1297 | 1313 | 1406 | 1313 | 1391 | 1390 | 1391 | 1437 | 1391 | 1361 | 51.4256745 |
| 65536000 | 2562 | 2594 | 2626 | 2812 | 2704 | 2734 | 2875 | 2984 | 2781 | 2718 | 2739 | 123.461735 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 thread |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 109 | 31 | 31 | 32 | 31 | 31 | 30 | 46 | 31 | 31 | 40.3 | 23.3368807 |
| 2000 | 47 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 31 | 32.6 | 4.8 |
| 4000 | 46 | 31 | 31 | 31 | 32 | 16 | 31 | 32 | 16 | 16 | 28.2 | 9.07524104 |
| 8000 | 32 | 31 | 32 | 31 | 31 | 31 | 32 | 31 | 31 | 31 | 31.3 | 0.45825757 |
| 16000 | 15 | 31 | 31 | 31 | 31 | 31 | 15 | 31 | 31 | 31 | 27.8 | 6.4 |
| 32000 | 16 | 16 | 31 | 32 | 31 | 16 | 32 | 31 | 31 | 32 | 26.8 | 7.08237248 |
| 64000 | 31 | 15 | 16 | 31 | 32 | 15 | 31 | 32 | 32 | 31 | 26.6 | 7.39188745 |
| 128000 | 31 | 47 | 47 | 31 | 31 | 31 | 31 | 46 | 31 | 31 | 35.7 | 7.18401002 |
| 256000 | 31 | 32 | 31 | 47 | 47 | 32 | 31 | 31 | 47 | 31 | 36 | 7.21110255 |
| 512000 | 47 | 46 | 47 | 63 | 46 | 46 | 47 | 47 | 47 | 47 | 48.3 | 4.92036584 |
| 1024000 | 94 | 78 | 77 | 78 | 94 | 63 | 77 | 78 | 62 | 78 | 77.9 | 9.9744674 |
| 2048000 | 125 | 140 | 110 | 110 | 125 | 141 | 125 | 110 | 110 | 108 | 120.4 | 12.0764233 |
| 4096000 | 219 | 235 | 204 | 202 | 234 | 219 | 218 | 203 | 203 | 219 | 215.6 | 11.8 |
| 8192000 | 391 | 391 | 407 | 391 | 407 | 391 | 406 | 375 | 376 | 423 | 395.8 | 14.1830885 |
| 16384000 | 703 | 736 | 734 | 796 | 781 | 766 | 781 | 798 | 782 | 797 | 767.4 | 30.764265 |
| 32768000 | 1359 | 1376 | 1359 | 1500 | 1358 | 1468 | 1485 | 1454 | 1469 | 1468 | 1429.6 | 55.7587661 |
| 65536000 | 2765 | 2766 | 2671 | 2874 | 2796 | 2812 | 2765 | 2983 | 2906 | 2813 | 2815.1 | 82.762854 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 32 | 46 | 31 | 47 | 47 | 32 | 32 | 48 | 31 | 30 | 37.6 | 7.70973411 |
| 2000 | 47 | 47 | 47 | 31 | 32 | 46 | 46 | 46 | 31 | 31 | 40.4 | 7.48598691 |
| 4000 | 31 | 31 | 31 | 46 | 46 | 31 | 47 | 47 | 47 | 47 | 40.4 | 7.68374908 |
| 8000 | 31 | 31 | 47 | 47 | 31 | 47 | 32 | 47 | 31 | 32 | 37.6 | 7.68374908 |
| 16000 | 47 | 47 | 31 | 32 | 47 | 31 | 46 | 47 | 32 | 31 | 39.1 | 7.71297608 |
| 32000 | 47 | 32 | 31 | 46 | 31 | 32 | 31 | 31 | 46 | 31 | 35.8 | 6.91086102 |
| 64000 | 31 | 31 | 32 | 47 | 31 | 47 | 47 | 47 | 31 | 31 | 37.5 | 7.76208735 |
| 128000 | 47 | 31 | 31 | 47 | 32 | 31 | 31 | 47 | 63 | 30 | 39 | 10.7424392 |
| 256000 | 46 | 47 | 46 | 47 | 47 | 47 | 32 | 62 | 47 | 47 | 46.8 | 6.72011905 |
| 512000 | 79 | 62 | 47 | 63 | 62 | 47 | 47 | 62 | 46 | 47 | 56.2 | 10.5337553 |
| 1024000 | 63 | 78 | 62 | 62 | 62 | 61 | 62 | 79 | 78 | 62 | 66.9 | 7.50266619 |
| 2048000 | 94 | 109 | 94 | 110 | 94 | 94 | 94 | 109 | 93 | 93 | **98.4** | 7.17216843 |
| 4096000 | 141 | 187 | 171 | 156 | 156 | 155 | 186 | 172 | 156 | 156 | 163.6 | 14.1081537 |
| 8192000 | 281 | 297 | 328 | 296 | 281 | 297 | 265 | 359 | 266 | 282 | 295.2 | 27.3780934 |
| 16384000 | 484 | 546 | 515 | 594 | 531 | 530 | 516 | 547 | 548 | 546 | 535.7 | 27.265546 |
| 32768000 | 953 | 1000 | 953 | 1063 | 999 | 1031 | 1079 | 1031 | 1000 | 953 | 1006.2 | 42.8667703 |
| 65536000 | 1718 | 1703 | 1797 | 2125 | 2032 | 2124 | 2125 | 2000 | 1970 | 1953 | 1954.7 | 154.654486 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 78 | 63 | 62 | 63 | 63 | 63 | 62 | 62 | 63 | 62 | 64.1 | 4.65725241 |
| 2000 | 78 | 47 | 63 | 77 | 62 | 61 | 63 | 63 | 78 | 79 | 67.1 | 9.9744674 |
| 4000 | 63 | 62 | 62 | 63 | 63 | 63 | 63 | 78 | 63 | 62 | 64.2 | 4.621688 |
| 8000 | 157 | 63 | 47 | 62 | 62 | 47 | 62 | 63 | 47 | 62 | 67.2 | 30.7109101 |
| 16000 | 62 | 63 | 62 | 63 | 62 | 62 | 62 | 78 | 62 | 63 | 63.9 | 4.72122865 |
| 32000 | 63 | 47 | 63 | 63 | 63 | 63 | 63 | 63 | 62 | 63 | 61.3 | 4.77598157 |
| 64000 | 62 | 62 | 47 | 62 | 63 | 63 | 62 | 62 | 63 | 62 | 60.8 | 4.621688 |
| 128000 | 62 | 62 | 62 | 78 | 62 | 62 | 79 | 62 | 62 | 63 | 65.4 | 6.56048779 |
| 256000 | 63 | 78 | 63 | 78 | 63 | 63 | 78 | 62 | 78 | 62 | 68.8 | 7.52063827 |
| 512000 | 62 | 63 | 77 | 78 | 78 | 78 | 77 | 79 | 79 | 62 | 73.3 | 7.21179589 |
| 1024000 | 78 | 78 | 95 | 79 | 94 | 77 | 94 | 93 | 94 | 78 | 86 | 8.02496106 |
| 2048000 | 110 | 109 | 109 | 109 | 93 | 93 | 108 | 125 | 109 | 110 | 107.5 | 8.65158945 |
| 4096000 | 172 | 173 | 156 | 172 | 172 | 156 | 141 | 171 | 156 | 156 | 162.5 | 10.413933 |
| 8192000 | 249 | 250 | 266 | 251 | 265 | 266 | 281 | 313 | 250 | 265 | 265.6 | 18.6021504 |
| 16384000 | 626 | 454 | 454 | 515 | 468 | 500 | 484 | 484 | 485 | 468 | 493.8 | 47.6797651 |
| 32768000 | 844 | 828 | 828 | 891 | 858 | 938 | 891 | 860 | 859 | 876 | 867.3 | 31.6829607 |
| 65536000 | 1609 | 1609 | 1641 | 1765 | 1609 | 1703 | 1719 | 1672 | 1671 | 1578 | 1657.6 | 56.0806562 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 125 | 110 | 124 | 126 | 124 | 110 | 125 | 110 | 140 | 125 | 121.9 | 8.98276127 |
| 2000 | 125 | 124 | 126 | 124 | 126 | 140 | 125 | 108 | 125 | 124 | 124.7 | 7.19791637 |
| 4000 | 125 | 110 | 140 | 141 | 125 | 110 | 126 | 110 | 110 | 125 | 122.2 | 11.4175304 |
| 8000 | 124 | 125 | 141 | 141 | 140 | 140 | 124 | 125 | 124 | 141 | 132.5 | 8.11480129 |
| 16000 | 125 | 109 | 141 | 125 | 141 | 109 | 109 | 109 | 110 | 125 | 120.3 | 12.4100766 |
| 32000 | 125 | 126 | 109 | 125 | 141 | 110 | 125 | 125 | 125 | 141 | 125.2 | 9.96794864 |
| 64000 | 110 | 108 | 109 | 125 | 125 | 110 | 125 | 110 | 125 | 124 | 117.1 | 7.72593036 |
| 128000 | 125 | 126 | 126 | 140 | 140 | 124 | 125 | 109 | 125 | 126 | 126.6 | 8.27284715 |
| 256000 | 141 | 124 | 124 | 141 | 125 | 126 | 141 | 125 | 125 | 124 | 129.6 | 7.48598691 |
| 512000 | 140 | 125 | 125 | 141 | 125 | 140 | 125 | 124 | 126 | 141 | 131.2 | 7.61314652 |
| 1024000 | 157 | 140 | 124 | 140 | 140 | 140 | 141 | 141 | 141 | 140 | 140.4 | 7.39188745 |
| 2048000 | 140 | 172 | 172 | 156 | 156 | 172 | 156 | 155 | 156 | 157 | 159.2 | 9.61041102 |
| 4096000 | 203 | 235 | 203 | 219 | 202 | 218 | 204 | 203 | 204 | 218 | 210.9 | 10.5304321 |
| 8192000 | 312 | 312 | 313 | 312 | 312 | 329 | 328 | 297 | 313 | 313 | 314.1 | 8.53756406 |
| 16384000 | 499 | 485 | 500 | 545 | 516 | 532 | 515 | 515 | 516 | 501 | 512.4 | 16.4328938 |
| 32768000 | 875 | 859 | 874 | 938 | 907 | 906 | 968 | 891 | 922 | 953 | 909.3 | 34.1234523 |
| 65536000 | 1766 | 1625 | 1656 | 1719 | 1671 | 1751 | 1750 | 1781 | 1703 | 1734 | 1715.6 | 48.4895865 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 16 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 265 | 218 | 250 | 265 | 219 | 234 | 267 | 265 | 250 | 249 | 248.2 | 17.8146008 |
| 2000 | 250 | 235 | 265 | 250 | 249 | 251 | 265 | 249 | 235 | 251 | 250 | 9.50789146 |
| 4000 | 250 | 235 | 266 | 266 | 267 | 218 | 266 | 267 | 265 | 265 | 256.5 | 16.1941347 |
| 8000 | 250 | 218 | 250 | 265 | 249 | 250 | 249 | 249 | 250 | 251 | 248.1 | 11.0313191 |
| 16000 | 265 | 235 | 249 | 250 | 250 | 219 | 266 | 266 | 266 | 250 | 251.6 | 14.6778745 |
| 32000 | 249 | 234 | 267 | 250 | 250 | 219 | 281 | 266 | 265 | 265 | 254.6 | 17.211624 |
| 64000 | 266 | 235 | 265 | 313 | 265 | 250 | 267 | 249 | 266 | 250 | 262.6 | 19.6428104 |
| 128000 | 266 | 234 | 249 | 250 | 250 | 234 | 264 | 250 | 250 | 281 | 252.8 | 13.5926451 |
| 256000 | 234 | 234 | 251 | 281 | 219 | 219 | 250 | 266 | 251 | 250 | 245.5 | 18.5755215 |
| 512000 | 234 | 234 | 264 | 265 | 235 | 219 | 250 | 250 | 250 | 250 | 245.1 | 13.7072973 |
| 1024000 | 281 | 250 | 282 | 250 | 250 | 265 | 249 | 266 | 250 | 281 | 262.4 | 13.7491818 |
| 2048000 | 265 | 265 | 296 | 281 | 265 | 265 | 265 | 282 | 265 | 312 | 276.1 | 15.7572206 |
| 4096000 | 313 | 312 | 328 | 313 | 313 | 313 | 328 | 328 | 313 | 327 | 318.8 | 7.31846979 |
| 8192000 | 407 | 391 | 406 | 437 | 406 | 453 | 422 | 421 | 406 | 406 | 415.5 | 17.2467388 |
| 16384000 | 625 | 610 | 593 | 656 | 610 | 625 | 688 | 625 | 640 | 609 | 628.1 | 26.0555176 |
| 32768000 | 985 | 985 | 1016 | 1031 | 1047 | 984 | 1094 | 1016 | 1016 | 999 | 1017.3 | 32.3853362 |
| 65536000 | 1749 | 1735 | 1750 | 1875 | 1797 | 1860 | 1859 | 1765 | 1844 | 1766 | 1800 | 51.3400429 |

**Experiment 4: Random Data Set: Ubuntu VM 2 Processors ( ms)**

**No threads**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0** | 0.1 | 0.3 |
| 2000 | 12 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1.3 | 3.579106034 |
| 4000 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0.3 |
| 8000 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0.3 |
| 16000 | 2 | 5 | 1 | 0 | 0 | 0 | 5 | 1 | 0 | 1 | 1.5 | 1.857417562 |
| 32000 | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 2 | 1 | 1 | 1.3 | 0.640312424 |
| 64000 | 7 | 2 | 3 | 4 | 2 | 2 | 3 | 2 | 2 | 2 | 2.9 | 1.513274595 |
| 128000 | 4 | 4 | 7 | 4 | 4 | 7 | 5 | 7 | 5 | 4 | 5.1 | 1.3 |
| 256000 | 9 | 12 | 9 | 9 | 12 | 16 | 8 | 10 | 10 | 14 | 10.9 | 2.42693222 |
| 512000 | 34 | 25 | 27 | 27 | 19 | 21 | 24 | 19 | 26 | 29 | 25.1 | 4.414748011 |
| 1024000 | 50 | 48 | 50 | 51 | 62 | 50 | 48 | 49 | 51 | 50 | 50.9 | 3.832753579 |
| 2048000 | 164 | 110 | 106 | 115 | 96 | 94 | 99 | 106 | 128 | 103 | 112.1 | 19.68476568 |
| 4096000 | 427 | 219 | 231 | 185 | 200 | 184 | 238 | 201 | 262 | 204 | 235.1 | 68.05652063 |
| 8192000 | 418 | 383 | 389 | 403 | 388 | 373 | 412 | 393 | 353 | 396 | 390.8 | 17.85385113 |
| 16384000 | 783 | 744 | 777 | 795 | 785 | 737 | 697 | 726 | 671 | 760 | 747.5 | 38.64776837 |
| 32768000 | 1486 | 1481 | 1526 | 1488 | 1540 | 1434 | 1406 | 1478 | 1507 | 1450 | 1479.6 | 38.78195457 |
| 65536000 | 3208 | 2864 | 3031 | 3092 | 3136 | 2794 | 2807 | 2988 | 2843 | 2765 | 2952.8 | 150.7002322 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 thread |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 42 | 21 | 16 | 14 | 25 | 16 | 31 | 18 | 28 | 19 | 23 | 8.23407554 |
| 2000 | 53 | 16 | 25 | 9 | 4 | 7 | 18 | 14 | 7 | 25 | 17.8 | 13.64404632 |
| 4000 | 36 | 27 | 6 | 6 | 5 | 14 | 33 | 15 | 19 | 18 | 17.9 | 10.54940757 |
| 8000 | 60 | 28 | 5 | 5 | 6 | 20 | 31 | 20 | 35 | 23 | 23.3 | 15.98780785 |
| 16000 | 48 | 16 | 18 | 12 | 12 | 6 | 19 | 6 | 16 | 25 | 17.8 | 11.47867588 |
| 32000 | 21 | 38 | 21 | 7 | 6 | 15 | 20 | 7 | 25 | 15 | 17.5 | 9.340770846 |
| 64000 | 33 | 40 | 8 | 10 | 7 | 27 | 13 | 14 | 24 | 17 | 19.3 | 10.65879918 |
| 128000 | 42 | 44 | 31 | 12 | 17 | 15 | 34 | 11 | 39 | 54 | 29.9 | 14.46685868 |
| 256000 | 36 | 57 | 22 | 21 | 20 | 36 | 27 | 32 | 55 | 49 | 35.5 | 13.20037878 |
| 512000 | 105 | 59 | 36 | 58 | 34 | 37 | 53 | 31 | 79 | 65 | 55.7 | 22.14068653 |
| 1024000 | 87 | 95 | 55 | 110 | 65 | 60 | 80 | 70 | 128 | 95 | 84.5 | 22.02385071 |
| 2048000 | 167 | 139 | 167 | 162 | 190 | 141 | 196 | 131 | 152 | 147 | 159.2 | 20.41470059 |
| 4096000 | 290 | 215 | 256 | 288 | 293 | 290 | 254 | 250 | 255 | 253 | 264.4 | 23.9382539 |
| 8192000 | 531 | 433 | 509 | 515 | 488 | 497 | 487 | 455 | 463 | 408 | 478.6 | 36.54640885 |
| 16384000 | 928 | 965 | 962 | 930 | 935 | 852 | 929 | 818 | 909 | 930 | 915.8 | 43.96771543 |
| 32768000 | 1691 | 1557 | 1712 | 1683 | 1761 | 1653 | 1695 | 1562 | 1753 | 1677 | 1674.4 | 65.38073111 |
| 65536000 | 3057 | 3218 | 3505 | 3113 | 3433 | 3050 | 3444 | 3128 | 3290 | 3317 | 3255.5 | 159.3306311 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 29 | 18 | 13 | 21 | 20 | 47 | 37 | 36 | 29 | 18 | 26.8 | 10.15677114 |
| 2000 | 20 | 12 | 24 | 30 | 25 | 25 | 23 | 13 | 37 | 19 | 22.8 | 7.068238819 |
| 4000 | 33 | 36 | 33 | 29 | 10 | 23 | 37 | 15 | 27 | 15 | 25.8 | 9.141115906 |
| 8000 | 29 | 9 | 35 | 10 | 29 | 32 | 40 | 23 | 20 | 32 | 25.9 | 9.782126558 |
| 16000 | 18 | 41 | 44 | 19 | 22 | 16 | 26 | 32 | 27 | 10 | 25.5 | 10.33682737 |
| 32000 | 30 | 10 | 27 | 28 | 24 | 84 | 14 | 19 | 29 | 34 | 29.9 | 19.38788281 |
| 64000 | 30 | 34 | 49 | 17 | 30 | 65 | 34 | 32 | 29 | 13 | 33.3 | 14.0431478 |
| 128000 | 62 | 20 | 47 | 33 | 37 | 60 | 22 | 36 | 19 | 36 | 37.2 | 14.52446212 |
| 256000 | 73 | 43 | 64 | 68 | 25 | 87 | 36 | 53 | 57 | 19 | 52.5 | 20.60218435 |
| 512000 | 79 | 33 | 108 | 79 | 64 | 45 | 51 | 59 | 44 | 34 | 59.6 | 22.37945486 |
| 1024000 | 122 | 75 | 118 | 147 | 71 | 80 | 120 | 101 | 138 | 108 | 108 | 24.8032256 |
| 2048000 | 171 | 189 | 222 | 145 | 149 | 221 | 143 | 202 | 186 | 227 | **185.5** | 30.96530316 |
| 4096000 | 276 | 266 | 288 | 251 | 283 | 291 | 282 | 241 | 331 | 273 | 278.2 | 23.21551206 |
| 8192000 | 460 | 457 | 401 | 489 | 502 | 466 | 411 | 426 | 427 | 432 | 447.1 | 31.45933884 |
| 16384000 | 767 | 714 | 767 | 771 | 721 | 706 | 724 | 664 | 689 | 755 | 727.8 | 34.61444785 |
| 32768000 | 1281 | 1336 | 1369 | 1419 | 1325 | 1268 | 1302 | 1396 | 1251 | 1297 | 1324.4 | 52.82461547 |
| 65536000 | 2500 | 2542 | 2427 | 2563 | 2542 | 2157 | 2492 | 2335 | 2905 | 2539 | 2500.2 | 179.80812 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 60 | 55 | 78 | 35 | 57 | 52 | 49 | 42 | 86 | 75 | 58.9 | 15.43016526 |
| 2000 | 32 | 35 | 36 | 53 | 70 | 51 | 66 | 89 | 31 | 65 | 52.8 | 18.54615863 |
| 4000 | 26 | 61 | 46 | 23 | 88 | 72 | 56 | 39 | 15 | 69 | 49.5 | 22.60641502 |
| 8000 | 54 | 104 | 41 | 31 | 82 | 92 | 93 | 60 | 91 | 73 | 72.1 | 23.31716106 |
| 16000 | 56 | 77 | 76 | 45 | 45 | 63 | 64 | 62 | 45 | 67 | 60 | 11.4629839 |
| 32000 | 66 | 70 | 61 | 21 | 48 | 62 | 54 | 59 | 43 | 89 | 57.3 | 17.02967997 |
| 64000 | 60 | 117 | 56 | 26 | 117 | 91 | 64 | 59 | 55 | 80 | 72.5 | 27.40529146 |
| 128000 | 119 | 87 | 93 | 71 | 48 | 51 | 70 | 101 | 78 | 45 | 76.3 | 23.06100605 |
| 256000 | 97 | 56 | 96 | 116 | 108 | 74 | 55 | 61 | 125 | 63 | 85.1 | 25.05374224 |
| 512000 | 140 | 141 | 137 | 91 | 100 | 134 | 117 | 139 | 127 | 74 | 120 | 22.58760722 |
| 1024000 | 170 | 100 | 179 | 116 | 137 | 147 | 183 | 181 | 126 | 128 | 146.7 | 28.4184799 |
| 2048000 | 209 | 221 | 164 | 238 | 212 | 172 | 308 | 194 | 214 | 193 | 212.5 | 38.17394399 |
| 4096000 | 329 | 322 | 361 | 355 | 342 | 267 | 347 | 297 | 345 | 289 | 325.4 | 29.72608282 |
| 8192000 | 440 | 478 | 512 | 520 | 500 | 466 | 459 | 432 | 446 | 515 | 476.8 | 31.3808859 |
| 16384000 | 681 | 692 | 726 | 824 | 696 | 646 | 866 | 714 | 821 | 703 | 736.9 | 69.42254677 |
| 32768000 | 1212 | 1301 | 1262 | 1450 | 1404 | 1145 | 1434 | 1348 | 1353 | 1341 | 1325 | 92.21171292 |
| 65536000 | 2491 | 2588 | 2607 | 2560 | 2704 | 2209 | 2737 | 2501 | 2446 | 2539 | 2538.2 | 139.7088401 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 76 | 100 | 66 | 81 | 92 | 124 | 151 | 142 | 119 | 115 | 106.6 | 26.92285275 |
| 2000 | 85 | 122 | 107 | 97 | 176 | 88 | 90 | 70 | 151 | 93 | 107.9 | 31.16552583 |
| 4000 | 126 | 108 | 64 | 132 | 94 | 187 | 112 | 84 | 125 | 166 | 119.8 | 34.82183223 |
| 8000 | 118 | 78 | 68 | 150 | 97 | 153 | 201 | 114 | 101 | 161 | 124.1 | 39.32035096 |
| 16000 | 148 | 111 | 120 | 155 | 123 | 61 | 124 | 140 | 84 | 117 | 118.3 | 26.96683148 |
| 32000 | 108 | 98 | 116 | 115 | 132 | 182 | 139 | 100 | 132 | 88 | 121 | 25.64371268 |
| 64000 | 169 | 103 | 89 | 140 | 121 | 115 | 140 | 108 | 157 | 130 | 127.2 | 23.60423691 |
| 128000 | 122 | 100 | 81 | 126 | 123 | 167 | 147 | 87 | 141 | 117 | 121.1 | 25.36710468 |
| 256000 | 125 | 109 | 159 | 163 | 82 | 111 | 145 | 108 | 113 | 140 | 125.5 | 24.4059419 |
| 512000 | 187 | 141 | 105 | 178 | 123 | 150 | 232 | 139 | 162 | 129 | 154.6 | 34.93765877 |
| 1024000 | 152 | 257 | 511 | 227 | 179 | 219 | 293 | 181 | 260 | 177 | 245.6 | 98.05631035 |
| 2048000 | 328 | 269 | 295 | 301 | 326 | 223 | 367 | 194 | 277 | 227 | 280.7 | 51.21923467 |
| 4096000 | 405 | 401 | 431 | 399 | 354 | 396 | 470 | 346 | 459 | 376 | 403.7 | 38.54361166 |
| 8192000 | 582 | 589 | 573 | 567 | 598 | 558 | 603 | 537 | 529 | 574 | 571 | 23.05645246 |
| 16384000 | 825 | 920 | 963 | 914 | 904 | 816 | 977 | 829 | 817 | 794 | 875.9 | 63.67801819 |
| 32768000 | 1349 | 1456 | 1524 | 1396 | 1462 | 1288 | 1460 | 1370 | 1383 | 1346 | 1403.4 | 67.16129838 |
| 65536000 | 2536 | 2794 | 2569 | 2523 | 2708 | 2303 | 2797 | 2420 | 2501 | 2601 | 2575.2 | 149.4508615 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 16 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| 1000 | 251 | 232 | 225 | 212 | 253 | 248 | 229 | 171 | 208 | 173 | 220.2 | 28.10978477 |
| 2000 | 190 | 274 | 222 | 267 | 222 | 258 | 212 | 191 | 183 | 237 | 225.6 | 31.18717685 |
| 4000 | 263 | 217 | 323 | 170 | 194 | 222 | 226 | 231 | 267 | 207 | 232 | 40.91698914 |
| 8000 | 255 | 286 | 296 | 222 | 180 | 266 | 198 | 191 | 188 | 169 | 225.1 | 44.45998201 |
| 16000 | 196 | 139 | 246 | 279 | 219 | 190 | 254 | 207 | 151 | 181 | 206.2 | 42.30555519 |
| 32000 | 290 | 179 | 243 | 234 | 219 | 213 | 269 | 192 | 239 | 209 | 228.7 | 32.0407553 |
| 64000 | 277 | 296 | 190 | 292 | 213 | 254 | 199 | 195 | 227 | 226 | 236.9 | 38.22159076 |
| 128000 | 575 | 246 | 247 | 284 | 271 | 268 | 297 | 190 | 304 | 189 | 287.1 | 103.0790473 |
| 256000 | 164 | 299 | 218 | 209 | 221 | 199 | 331 | 271 | 309 | 231 | 245.2 | 51.5340664 |
| 512000 | 230 | 313 | 338 | 260 | 213 | 246 | 254 | 233 | 298 | 286 | 267.1 | 38.18232575 |
| 1024000 | 351 | 460 | 367 | 274 | 304 | 284 | 332 | 322 | 330 | 251 | 327.5 | 55.57382477 |
| 2048000 | 460 | 302 | 396 | 355 | 405 | 360 | 382 | 387 | 410 | 459 | 391.6 | 44.98710926 |
| 4096000 | 573 | 442 | 601 | 529 | 592 | 567 | 500 | 514 | 437 | 562 | 531.7 | 55.36072615 |
| 8192000 | 732 | 856 | 795 | 802 | 745 | 749 | 825 | 627 | 747 | 763 | 764.1 | 59.37920511 |
| 16384000 | 981 | 1192 | 1160 | 1129 | 1050 | 947 | 1231 | 966 | 1013 | 1042 | 1071.1 | 95.22231881 |
| 32768000 | 1401 | 1683 | 1781 | 1733 | 1476 | 1248 | 1772 | 1452 | 1449 | 1451 | 1544.6 | 173.7867659 |
| 65536000 | 2595 | 3144 | 2866 | 2855 | 2751 | 2421 | 3085 | 2398 | 2454 | 2685 | 2725.4 | 251.7741845 |

**Experiment 5:Random Data Set: Windows 10 Laptop (ms)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.2 | 0.6 |
| 2000 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.1 | 0.3 |
| 4000 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0.5 | 0.5 |
| 8000 | 3 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0.5 | 0.92195445 |
| 16000 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0.6 | 0.48989795 |
| 32000 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.9 | 0.3 |
| 64000 | 3 | 2 | 0 | 2 | 2 | 3 | 3 | 3 | 2 | 2 | 2.2 | 0.87177979 |
| 128000 | 4 | 4 | 0 | 4 | 5 | 4 | 5 | 5 | 5 | 5 | 4.1 | 1.44568323 |
| 256000 | 9 | 8 | 22 | 9 | 9 | 8 | 10 | 9 | 8 | 8 | 10 | 4.04969135 |
| 512000 | 17 | 17 | 16 | 17 | 17 | 17 | 22 | 17 | 18 | 19 | 17.7 | 1.61554944 |
| 1024000 | 37 | 36 | 31 | 39 | 42 | 39 | 48 | 41 | 42 | 39 | 39.4 | 4.22374242 |
| 2048000 | 81 | 79 | 63 | 79 | 80 | 79 | 85 | 81 | 79 | 78 | 78.4 | 5.46260011 |
| 4096000 | 162 | 158 | 147 | 152 | 153 | 151 | 161 | 155 | 150 | 161 | 155 | 4.97995984 |
| 8192000 | 314 | 312 | 300 | 308 | 312 | 307 | 352 | 307 | 311 | 307 | 313 | 13.5277493 |
| 16384000 | 633 | 652 | 601 | 606 | 606 | 612 | 627 | 603 | 621 | 628 | 618.9 | 15.4948379 |
| 32768000 | 1265 | 1263 | 1184 | 1205 | 1216 | 1228 | 1222 | 1214 | 1246 | 1235 | 1227.8 | 24.0989626 |
| 65536000 | 2502 | 2493 | 2450 | 2451 | 2456 | 2454 | 2523 | 2458 | 2488 | 2537 | 2481.2 | 30.459153 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run/elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 20 | 18 | 15 | 14 | 15 | 13 | 12 | 15 | 15 | 15 | 15.2 | 2.18174242 |
| 2000 | 19 | 11 | 0 | 11 | 10 | 11 | 11 | 12 | 11 | 11 | 10.7 | 4.31393092 |
| 4000 | 14 | 12 | 16 | 11 | 12 | 11 | 10 | 11 | 11 | 11 | 11.9 | 1.7 |
| 8000 | 13 | 11 | 16 | 11 | 12 | 11 | 11 | 11 | 11 | 12 | 11.9 | 1.5132746 |
| 16000 | 17 | 13 | 15 | 11 | 12 | 12 | 13 | 12 | 12 | 11 | 12.8 | 1.77763888 |
| 32000 | 14 | 12 | 14 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12.4 | 0.8 |
| 64000 | 14 | 14 | 14 | 13 | 13 | 13 | 13 | 13 | 13 | 13 | 13.3 | 0.45825757 |
| 128000 | 18 | 16 | 6 | 16 | 16 | 16 | 15 | 16 | 17 | 15 | 15.1 | 3.14483704 |
| 256000 | 26 | 21 | 32 | 26 | 24 | 24 | 25 | 24 | 25 | 24 | 25.1 | 2.66270539 |
| 512000 | 46 | 47 | 46 | 43 | 40 | 48 | 47 | 41 | 39 | 46 | 44.3 | 3.1 |
| 1024000 | 79 | 58 | 68 | 58 | 60 | 59 | 58 | 60 | 56 | 62 | 61.8 | 6.52380257 |
| 2048000 | 122 | 98 | 116 | 106 | 122 | 109 | 117 | 120 | 118 | 119 | 114.7 | 7.47060908 |
| 4096000 | 203 | 180 | 201 | 204 | 202 | 218 | 198 | 202 | 203 | 211 | 202.2 | 9.18477 |
| 8192000 | 354 | 369 | 378 | 357 | 351 | 334 | 354 | 342 | 367 | 338 | 354.4 | 13.3656276 |
| 16384000 | 665 | 771 | 670 | 642 | 654 | 666 | 670 | 649 | 653 | 679 | 671.9 | 34.7230471 |
| 32768000 | 1298 | 1437 | 1301 | 1260 | 1270 | 1274 | 1260 | 1250 | 1289 | 1334 | 1297.3 | 52.2054595 |
| 65536000 | 2509 | 2550 | 2549 | 2504 | 2597 | 2504 | 2486 | 2489 | 2486 | 2608 | 2528.2 | 43.182867 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 22 | 21 | 20 | 22 | 21 | 20 | 19 | 20 | 19 | 19 | 20.3 | 1.1 |
| 2000 | 24 | 18 | 15 | 16 | 19 | 19 | 22 | 17 | 24 | 22 | 19.6 | 3.0724583 |
| 4000 | 18 | 19 | 16 | 16 | 21 | 20 | 16 | 16 | 21 | 16 | 17.9 | 2.07123152 |
| 8000 | 18 | 18 | 31 | 16 | 17 | 20 | 17 | 16 | 20 | 16 | 18.9 | 4.27668096 |
| 16000 | 17 | 19 | 22 | 16 | 16 | 20 | 17 | 16 | 19 | 17 | 17.9 | 1.92093727 |
| 32000 | 17 | 18 | 16 | 16 | 16 | 21 | 21 | 17 | 20 | 18 | 18 | 1.8973666 |
| 64000 | 18 | 19 | 15 | 17 | 17 | 19 | 21 | 17 | 20 | 25 | 18.8 | 2.63818119 |
| 128000 | 18 | 20 | 16 | 21 | 18 | 18 | 27 | 17 | 22 | 23 | 20 | 3.16227766 |
| 256000 | 23 | 21 | 16 | 27 | 29 | 20 | 27 | 27 | 29 | 25 | 24.4 | 4.07921561 |
| 512000 | 31 | 32 | 22 | 37 | 39 | 32 | 39 | 38 | 39 | 33 | 34.2 | 5.11468474 |
| 1024000 | 40 | 37 | 47 | 42 | 43 | 44 | 43 | 45 | 44 | 44 | 42.9 | 2.62488095 |
| 2048000 | 63 | 69 | 66 | 58 | 60 | 60 | 60 | 71 | 58 | 58 | **62.3** | 4.53982379 |
| 4096000 | 123 | 139 | 100 | 105 | 106 | 117 | 103 | 104 | 108 | 109 | 111.4 | 11.2712022 |
| 8192000 | 206 | 223 | 200 | 198 | 208 | 214 | 214 | 212 | 204 | 228 | 210.7 | 9.07799537 |
| 16384000 | 390 | 397 | 384 | 376 | 379 | 382 | 377 | 378 | 382 | 376 | 382.1 | 6.44127317 |
| 32768000 | 750 | 849 | 770 | 728 | 724 | 731 | 732 | 727 | 722 | 773 | 750.6 | 37.2617767 |
| 65536000 | 1472 | 1576 | 1471 | 1453 | 1459 | 1459 | 1444 | 1462 | 1449 | 1511 | 1475.6 | 37.8423044 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 37 | 30 | 21 | 29 | 30 | 33 | 28 | 34 | 29 | 31 | 30.2 | 4.01995025 |
| 2000 | 63 | 29 | 32 | 28 | 28 | 30 | 29 | 32 | 28 | 30 | 32.9 | 10.1336075 |
| 4000 | 37 | 29 | 31 | 28 | 29 | 29 | 29 | 29 | 28 | 30 | 29.9 | 2.50798724 |
| 8000 | 32 | 30 | 37 | 28 | 28 | 29 | 28 | 28 | 28 | 28 | 29.6 | 2.76405499 |
| 16000 | 31 | 31 | 16 | 31 | 28 | 29 | 28 | 28 | 28 | 28 | 27.8 | 4.14246304 |
| 32000 | 37 | 29 | 31 | 27 | 28 | 28 | 28 | 29 | 28 | 28 | 29.3 | 2.75862284 |
| 64000 | 32 | 31 | 31 | 29 | 29 | 32 | 30 | 30 | 28 | 29 | 30.1 | 1.3 |
| 128000 | 31 | 31 | 38 | 31 | 33 | 30 | 31 | 31 | 32 | 30 | 31.8 | 2.22710575 |
| 256000 | 37 | 31 | 31 | 31 | 32 | 30 | 32 | 31 | 30 | 30 | 31.5 | 1.96214169 |
| 512000 | 32 | 34 | 37 | 33 | 34 | 33 | 33 | 35 | 35 | 33 | 33.9 | 1.37477271 |
| 1024000 | 51 | 40 | 47 | 42 | 46 | 42 | 43 | 47 | 42 | 44 | 44.4 | 3.13687743 |
| 2048000 | 63 | 64 | 82 | 62 | 60 | 60 | 60 | 61 | 64 | 61 | 63.7 | 6.27773845 |
| 4096000 | 101 | 104 | 116 | 101 | 108 | 99 | 104 | 103 | 101 | 103 | 104 | 4.6260134 |
| 8192000 | 185 | 225 | 216 | 186 | 188 | 181 | 189 | 188 | 212 | 187 | 195.7 | 14.8327341 |
| 16384000 | 353 | 362 | 354 | 482 | 341 | 348 | 343 | 340 | 342 | 367 | 363.2 | 40.5433102 |
| 32768000 | 669 | 766 | 701 | 654 | 652 | 655 | 651 | 654 | 654 | 687 | 674.3 | 34.5833775 |
| 65536000 | 1303 | 1279 | 1332 | 1283 | 1281 | 1279 | 1277 | 1279 | 1272 | 1277 | 1286.2 | 17.1569228 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 54 | 55 | 55 | 55 | 55 | 54 | 55 | 55 | 55 | 55 | 54.8 | 0.4 |
| 2000 | 62 | 57 | 57 | 57 | 54 | 55 | 56 | 56 | 55 | 55 | 56.4 | 2.10713075 |
| 4000 | 53 | 56 | 56 | 54 | 56 | 56 | 56 | 55 | 56 | 55 | 55.3 | 1.00498756 |
| 8000 | 63 | 57 | 60 | 57 | 57 | 57 | 56 | 59 | 57 | 57 | 58 | 2 |
| 16000 | 53 | 55 | 56 | 57 | 55 | 56 | 56 | 55 | 55 | 56 | 55.4 | 1.0198039 |
| 32000 | 47 | 55 | 55 | 55 | 55 | 56 | 55 | 55 | 57 | 56 | 54.6 | 2.61533937 |
| 64000 | 69 | 56 | 56 | 56 | 56 | 59 | 57 | 55 | 56 | 56 | 57.6 | 3.92937654 |
| 128000 | 53 | 61 | 57 | 59 | 59 | 60 | 59 | 58 | 57 | 57 | 58 | 2.0976177 |
| 256000 | 63 | 58 | 60 | 57 | 59 | 57 | 58 | 57 | 60 | 58 | 58.7 | 1.79164729 |
| 512000 | 69 | 61 | 63 | 64 | 64 | 63 | 62 | 63 | 63 | 64 | 63.6 | 2.00997512 |
| 1024000 | 69 | 71 | 119 | 75 | 74 | 76 | 73 | 72 | 72 | 73 | 77.4 | 13.9942845 |
| 2048000 | 84 | 86 | 92 | 83 | 86 | 83 | 82 | 84 | 84 | 82 | 84.6 | 2.8 |
| 4096000 | 122 | 121 | 120 | 124 | 121 | 116 | 118 | 118 | 119 | 120 | 119.9 | 2.16564078 |
| 8192000 | 200 | 199 | 202 | 209 | 201 | 197 | 197 | 199 | 200 | 201 | 200.5 | 3.23264598 |
| 16384000 | 369 | 353 | 360 | 352 | 353 | 354 | 351 | 350 | 353 | 356 | 355.1 | 5.33760246 |
| 32768000 | 686 | 663 | 678 | 661 | 661 | 663 | 661 | 662 | 661 | 661 | 665.7 | 8.37914077 |
| 65536000 | 1311 | 1291 | 1667 | 1286 | 1288 | 1291 | 1284 | 1286 | 1285 | 1285 | 1327.4 | 113.448843 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 16 threads |  |  |  |  |  |  |  |  |  |  |  |  |
| Run  elements | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Average | Standard Dev |
| 1000 | 103 | 110 | 115 | 108 | 107 | 106 | 108 | 108 | 112 | 108 | 108.5 | 3.10644491 |
| 2000 | 122 | 110 | 110 | 109 | 109 | 108 | 109 | 111 | 111 | 112 | 111.1 | 3.80657326 |
| 4000 | 118 | 109 | 115 | 149 | 148 | 150 | 150 | 149 | 147 | 147 | 138.2 | 16.0049992 |
| 8000 | 119 | 111 | 112 | 113 | 113 | 112 | 114 | 113 | 112 | 111 | 113 | 2.19089023 |
| 16000 | 104 | 127 | 110 | 111 | 111 | 113 | 112 | 110 | 111 | 112 | 112.1 | 5.48543526 |
| 32000 | 121 | 203 | 110 | 112 | 113 | 112 | 112 | 112 | 112 | 112 | 121.9 | 27.1788521 |
| 64000 | 111 | 111 | 112 | 113 | 112 | 110 | 111 | 112 | 111 | 112 | 111.5 | 0.80622577 |
| 128000 | 101 | 118 | 111 | 114 | 114 | 114 | 114 | 114 | 113 | 114 | 112.7 | 4.22018957 |
| 256000 | 116 | 113 | 113 | 112 | 112 | 113 | 113 | 113 | 113 | 113 | 113.1 | 1.04403065 |
| 512000 | 122 | 119 | 119 | 120 | 118 | 116 | 118 | 117 | 119 | 117 | 118.5 | 1.62788206 |
| 1024000 | 118 | 122 | 123 | 117 | 116 | 117 | 119 | 118 | 116 | 116 | 118.2 | 2.35796522 |
| 2048000 | 132 | 137 | 133 | 134 | 133 | 132 | 132 | 134 | 134 | 135 | 133.6 | 1.49666295 |
| 4096000 | 166 | 164 | 163 | 160 | 161 | 163 | 157 | 162 | 160 | 160 | 161.6 | 2.41660919 |
| 8192000 | 231 | 244 | 486 | 244 | 241 | 240 | 243 | 248 | 241 | 241 | 265.9 | 73.482583 |
| 16384000 | 386 | 389 | 488 | 383 | 383 | 381 | 382 | 383 | 383 | 378 | 393.6 | 31.5854397 |
| 32768000 | 702 | 689 | 729 | 691 | 699 | 688 | 690 | 687 | 691 | 691 | 695.7 | 11.9920807 |
| 65536000 | 1323 | 1314 | 1384 | 1308 | 1400 | 1306 | 1303 | 1307 | 1307 | 1306 | 1325.8 | 33.7158716 |

**Data Set Average Times (ms)**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No threads:  elements | Random | Discrete | Uniform | Bernoulli | Gaussian | Zero | Standard  Deviation |
| 1000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4000 | 1.5 | 0 | 0 | 0 | 0 | 0 | 0.559017 |
| 8000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16000 | 3 | 0 | 0 | 1.6 | 1.5 | 3.3 | 1.289272 |
| 32000 | 1.5 | 3 | 0 | 3.3 | 0 | 0 | 1.414214 |
| 64000 | 0 | 1.6 | 0 | 3.2 | 1.6 | 4.6 | 1.650926 |
| 128000 | 3.1 | 3.2 | 3.2 | 4.7 | 8.1 | 1.5 | 2.066935 |
| 256000 | 7.9 | 4.9 | 10.9 | 7.8 | 9.4 | 10.9 | 2.083 |
| 512000 | 22 | 18.8 | 20.6 | 19.1 | 17.4 | 19 | 1.458786 |
| 1024000 | 42.3 | 41.8 | 40.2 | 41.1 | 36.1 | 34.2 | 3.042705 |
| 2048000 | 85.6 | 82.8 | 79.7 | 82.9 | 79.4 | 76.6 | 2.928405 |
| 4096000 | 168.8 | 165.8 | 163.8 | 171.9 | 164.3 | 162.5 | 3.226668 |
| 8192000 | 336.1 | 337.4 | 337.4 | 332.8 | 332.7 | 329.6 | 2.868023 |
| 16384000 | 688.9 | 690.9 | 690.6 | 681 | 684.5 | 698.4 | 5.458556 |
| 32768000 | 1361 | 1351.2 | 1351.8 | 1353 | 1362.6 | 1359.7 | 4.656805 |
| 65536000 | 2739 | 2711 | 2715.9 | 2698.4 | 2717.7 | 2734.5 | 13.77721 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 1 Thread  elements | Random | Discrete | Uniform | Bernoulli | Gaussian | Zero | Standard  Deviation |
| 1000 | 40.3 | 34.1 | 34 | 35.7 | 34.2 | 32.2 | 2.543892 |
| 2000 | 32.6 | 31.3 | 32.9 | 26.5 | 31.1 | 29.7 | 2.145085 |
| 4000 | 28.2 | 28 | 28 | 26.5 | 31.4 | 28.2 | 1.472432 |
| 8000 | 31.3 | 23.4 | 31.4 | 31.4 | 31.1 | 26.4 | 3.140418 |
| 16000 | 27.8 | 34.3 | 31.1 | 26.5 | 33 | 28.1 | 2.866279 |
| 32000 | 26.8 | 27.9 | 29.7 | 31.3 | 31.1 | 26.8 | 1.873203 |
| 64000 | 26.6 | 28.1 | 32.7 | 29.8 | 32.7 | 28 | 2.347161 |
| 128000 | 35.7 | 32.6 | 36 | 31.1 | 32.7 | 37.5 | 2.264705 |
| 256000 | 36 | 38.8 | 40.6 | 37.4 | 42.2 | 36 | 2.30579 |
| 512000 | 48.3 | 51.7 | 53.3 | 54.4 | 57.5 | 51.6 | 2.822528 |
| 1024000 | 77.9 | 76.5 | 71.9 | 78 | 88.8 | 72.2 | 5.605578 |
| 2048000 | 120.4 | 120.3 | 123.2 | 121.6 | 128 | 116.9 | 3.382143 |
| 4096000 | 215.6 | 217.2 | 215.6 | 212.4 | 221.9 | 204.9 | 5.179768 |
| 8192000 | 395.8 | 387.5 | 392.4 | 385.8 | 393.7 | 392.2 | 3.480741 |
| 16384000 | 767.4 | 764.2 | 756.1 | 751.3 | 758.2 | 768.6 | 6.256907 |
| 32768000 | 1429.6 | 1431.3 | 1429.6 | 1436.1 | 1434.5 | 1448.4 | 6.493437 |
| 65536000 | 2815.1 | 2779.7 | 2759.3 | 2760.8 | 2720.2 | 2787.3 | 29.17076 |
| 2 threads:  elements | Random | Discrete | Uniform | Bernoulli | Gaussian | Zero | Standard  Deviation |
| 1000 | 37.6 | 41.8 | 42.2 | 38.5 | 40.2 | 39 | 1.685642 |
| 2000 | 40.4 | 34.4 | 40.4 | 42.2 | 36.1 | 42.1 | 2.969661 |
| 4000 | 40.4 | 42.2 | 39.3 | 37.6 | 35.9 | 37.4 | 2.088859 |
| 8000 | 37.6 | 35.8 | 36 | 32.5 | 36.1 | 34.4 | 1.594783 |
| 16000 | 39.1 | 40.6 | 35.9 | 40.7 | 33.1 | 36.2 | 2.767671 |
| 32000 | 35.8 | 32.8 | 38.9 | 35.8 | 37.5 | 39.2 | 2.181488 |
| 64000 | 37.5 | 40.5 | 34.4 | 39.2 | 40.4 | 35.9 | 2.275534 |
| 128000 | 39 | 37 | 39.1 | 34.2 | 40.4 | 40.7 | 2.226357 |
| 256000 | 46.8 | 42.4 | 38.6 | 39.1 | 43.5 | 40.5 | 2.812719 |
| 512000 | 56.2 | 51.4 | 47 | 51.3 | 51 | 51.7 | 2.66625 |
| 1024000 | 66.9 | 65.8 | 70.1 | 67.2 | 69.9 | 66.9 | 1.616581 |
| 2048000 | 98.4 | 99.7 | 95.2 | 93.7 | 101.5 | 96.8 | 2.64118 |
| 4096000 | 163.6 | 159.8 | 165.4 | 162.3 | 162.6 | 153.3 | 3.893014 |
| 8192000 | 295.2 | 282.6 | 292.1 | 287.5 | 284.4 | 281.5 | 4.993134 |
| 16384000 | 535.7 | 534.5 | 532.7 | 546.9 | 531.3 | 535.9 | 5.065789 |
| 32768000 | 1006.2 | 1027.9 | 1001.7 | 1004.8 | 1009.4 | 999.8 | 9.290138 |
| 65536000 | 1954.7 | 1892.2 | 1859.2 | 1851.8 | 1829.6 | 1934.7 | 45.07818 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 4 Thread  elements | Random | Discrete | Uniform | Bernoulli | Gaussian | Zero | Standard  Deviation |
| 1000 | 64.1 | 65.5 | 63.9 | 62.5 | 60.7 | 64.2 | 1.519229 |
| 2000 | 67.1 | 62.3 | 64.2 | 60.8 | 56.2 | 62.5 | 3.318341 |
| 4000 | 64.2 | 64.2 | 59.3 | 54.5 | 58.1 | 59.3 | 3.417927 |
| 8000 | 67.2 | 60.8 | 64.2 | 64.4 | 64.2 | 60.8 | 2.236068 |
| 16000 | 63.9 | 62.7 | 56.2 | 59.5 | 60.7 | 61.1 | 2.452493 |
| 32000 | 61.3 | 66.9 | 64.2 | 60.7 | 56.2 | 64.3 | 3.404246 |
| 64000 | 60.8 | 65.8 | 59.3 | 56.2 | 62.4 | 60.8 | 2.913427 |
| 128000 | 65.4 | 67.5 | 61 | 60.9 | 59.1 | 65.7 | 3.068478 |
| 256000 | 68.8 | 70 | 65.4 | 68.7 | 67.3 | 65.4 | 1.74069 |
| 512000 | 73.3 | 73.5 | 74.6 | 68.1 | 74.6 | 72.1 | 2.226357 |
| 1024000 | 86 | 90.6 | 89.1 | 84.4 | 80.8 | 84.2 | 3.252563 |
| 2048000 | 107.5 | 109.2 | 109.1 | 106.3 | 107.2 | 109.5 | 1.195361 |
| 4096000 | 162.5 | 159.3 | 159.1 | 159.4 | 157.4 | 164 | 2.243076 |
| 8192000 | 265.6 | 207.75 | 256.5 | 257.4 | 259.3 | 262.4 | 19.80039 |
| 16384000 | 493.8 | 484.5 | 470.3 | 470 | 468.9 | 482.8 | 9.310821 |
| 32768000 | 867.3 | 867.3 | 864.3 | 850.1 | 856 | 882.4 | 10.1301 |
| 65536000 | 1657.6 | 1645.3 | 1640.3 | 1638.9 | 1589 | 1664.1 | 24.21129 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 8 threads:  elements | Random | Discrete | Uniform | Bernoulli | Gaussian | Zero | Standard  Deviation |
| 1000 | 121.9 | 120.2 | 119.8 | 121.7 | 118.5 | 123.2 | 1.548566 |
| 2000 | 124.7 | 118.7 | 122.2 | 123.3 | 118.5 | 118.8 | 2.476332 |
| 4000 | 122.2 | 122 | 126.5 | 118.6 | 112.3 | 120.2 | 4.317407 |
| 8000 | 132.5 | 124.9 | 120.4 | 126.9 | 119 | 123.5 | 4.43421 |
| 16000 | 120.3 | 120.4 | 126.4 | 117.3 | 121.8 | 118.7 | 2.870201 |
| 32000 | 125.2 | 123.4 | 118.7 | 120.2 | 117 | 125.2 | 3.180365 |
| 64000 | 117.1 | 121.9 | 115.7 | 117.3 | 118.6 | 120.5 | 2.113778 |
| 128000 | 126.6 | 128.1 | 120.3 | 123.2 | 121.5 | 126.1 | 2.829016 |
| 256000 | 129.6 | 128.1 | 122 | 123.4 | 117.4 | 126.8 | 4.125429 |
| 512000 | 131.2 | 132.6 | 125 | 131 | 124.9 | 126.5 | 3.150485 |
| 1024000 | 140.4 | 138.8 | 138.7 | 144.9 | 145.2 | 139.1 | 2.791306 |
| 2048000 | 159.2 | 164 | 160.9 | 160.7 | 164 | 170.2 | 3.601234 |
| 4096000 | 210.9 | 214 | 209.2 | 206.2 | 204.4 | 202.9 | 3.830434 |
| 8192000 | 314.1 | 309.2 | 312.6 | 306.5 | 304.4 | 307.7 | 3.368193 |
| 16384000 | 512.4 | 513.8 | 548.4 | 515.5 | 508 | 521.6 | 13.35137 |
| 32768000 | 909.3 | 902.9 | 904.5 | 899.9 | 901.1 | 926.5 | 9.064706 |
| 65536000 | 1715.6 | 1695.1 | 1735.7 | 1690.1 | 1670.5 | 1717.2 | 21.24599 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 16 Thread  elements | Random | Discrete | Uniform | Bernoulli | Gaussian | Zero | Standard  Deviation |
| 1000 | 248.2 | 241.9 | 239.1 | 242 | 240.6 | 239 | 3.097849 |
| 2000 | 250 | 248.7 | 239 | 249.9 | 235.9 | 246.9 | 5.554178 |
| 4000 | 256.5 | 249.9 | 248.5 | 247 | 243.6 | 254.8 | 4.425212 |
| 8000 | 248.1 | 253 | 268.7 | 256 | 242.2 | 251.4 | 8.148347 |
| 16000 | 251.6 | 250.2 | 248.5 | 259.5 | 245.1 | 253.4 | 4.456986 |
| 32000 | 254.6 | 251.2 | 251.6 | 254.8 | 239 | 252.9 | 5.397659 |
| 64000 | 262.6 | 248.5 | 251.6 | 250 | 237.6 | 247 | 7.361103 |
| 128000 | 252.8 | 242.4 | 265.6 | 238.9 | 241.9 | 248.4 | 8.978431 |
| 256000 | 245.5 | 245.4 | 249.9 | 246.9 | 245.2 | 257.7 | 4.446597 |
| 512000 | 245.1 | 248.1 | 256.3 | 243.5 | 247 | 259.5 | 5.898705 |
| 1024000 | 262.4 | 259.1 | 250 | 251.5 | 254.2 | 273.4 | 7.929831 |
| 2048000 | 276.1 | 276.5 | 282.8 | 276.6 | 270 | 287.3 | 5.493456 |
| 4096000 | 318.8 | 323.2 | 337.1 | 329.8 | 318.5 | 332.6 | 7.005633 |
| 8192000 | 415.5 | 409.1 | 414.2 | 418.5 | 421.4 | 407.8 | 4.808124 |
| 16384000 | 628.1 | 611 | 618.8 | 621.9 | 640.5 | 616.8 | 9.428812 |
| 32768000 | 1017.3 | 996.9 | 1015.5 | 1029.8 | 1040.8 | 1012.2 | 13.79188 |
| 65536000 | 1800 | 1792.4 | 1798.1 | 1845.6 | 1740.4 | 1820.5 | 31.91562 |