CPSC 131 Final Exam Analysis Report

|  |  |
| --- | --- |
| **Name:** | Huy Gia |
| **CWID:** | 893576058 |
| **Section:** | 04 |
| **Instructor:** | T Bettens |

Table of Contents

[Part A Inserting into a Vector at the back versus a Hash Table 2](#_Toc102986972)

[Part B Removing from a SLL at the back versus a Vector at the front 3](#_Toc102986973)

[Part C Searching a Vector, DLL, SLL, BST, and a Hash Table 4](#_Toc102986974)

[Part D {You select #1} 5](#_Toc102986975)

[Part E {You select #2} 6](#_Toc102986976)

Instructions:

* Use this template to prepare your Final Exam response.
* Replace and remove the placeholders (everything in between and including the curly braces) with your specific information. **Remove all placeholders before submitting your exam**.
* Remove unused table rows
* Document formatting, such as margins, font, font size, paragraph spacing, etc., must not be altered.

Note: In MSWord, F9 will (usually) update references in selected text, such as figure numbers or the table of contents.

# Inserting into a Vector at the back versus a Hash Table

## Data Analysis Graphs

|  |  |
| --- | --- |
|  |  |
| Figure ‑: {Insert at the back - Vector} | Figure ‑: {Insert – Hash Table} |
|  |  |
| Figure ‑: {Hash Table vs Vector insert at the back} | Figure ‑: { Hash Table vs Vector insert at the back } |

## Analysis and Summation

{Explain the information in the graph in terms of the operation’s efficiency class (Big-Oh). Talk to what you see on the graphs and explain why you see it. If the graphs are not what you expected, point that out and explain. Compare and contrast the operations performed on the different data structures. Identify and describe patterns in the graphs, including peaks, trends, and variability.}

{Do not indent or place blank lines between paragraphs.}

* In this analysis, we are comparing between inserting into a vector at the back versus inserting into a hash table. As you can see the hash table graph on the upper right, with the size of 10000, we can see that it runs as a O(1) time when we are inserting using a hash table. A hash tables is a data structure that maps keys to their associated values using a hash function. Looking at the graph on the bottom left when I compared between hash table versus vector inserting at the back , we can see that both data structure running in the O(1) times. Which means both of them are really efficiency in this inserting function. Then, looking at the graph on the bottom right, when I compared both of them but with larger size of data, we can see the changing between hash table and vector. Hash tables seem more consistency comparing to vector, the blue line of the vector seems go way too high in some parts of the data elements even though both run in constant time. As we already known, hash table values are not stored contiguously, it will assign each key to a unique bucket. On the other hand, vector holds an ordered collection of elements. It is essentially a dynamic array. If we neglect the size of element, both of these data structure are two of the most useful aspects in the area. However, a hash tables seems way faster than a vector. If you take a look at both of the graphs on the bottom left and right, you can clearly saw the differences. The graph on the bottom left with the size around 10000 elements, we can see how the hash table line (orange) looks similar to the straight line compared to the vector line (blue). We can see that vector lines go up and down and not really straight. Both of them run around 0.0002 seconds but the vector sometimes go up to 0.0004 or 0.0005 seconds. When I do the calculation, for the size of 10,000 elements, a hash table is 85.4 times faster than a vector. Secondly, when I compare both of them with a size of 90000 elements, we can see a big differences on the bottom right graph. A hash tables lines seem not change a lot comparing to vector line. A hash table still really consistency, only some of the line go up but not much. However, on the vector graph, we can see the changing in here. It seems to go up to 2 seconds in some of the data elements. With a small size of data elements, vector seem to be more efficiency and cheaper. But as the data structure increases in size, hash table is really efficiency in insert and stores elements. In conclusion, a hash table function performs better than the vector even though both of them are running at O(1) times. The size of the data elements is really important between those function. With a small size, they seem both the same when we look at the graph on the bottom left. They are not much different with 10,000 sizes of elements. Hash table seem more straight line comparing to vector line. Since hash table access the items in a hash table by a key rather than a position. On the other hand, vector holds an ordered collection of elements. When I look at both of the graph, it is exactly what I expected and understand why both of them performs like that on the graph. Their patterns seem the same in the small size of data elements, but we can see more clearly when I increased the size into around 100,000 elements. In the large size of data elements, a hash table is outstanding comparing to vector. They both running in 0(1) time, their peaks, and trends similar to the straight line. Hash tables look more straight rather than vector. Their variability seems more clearly when I adjust the size of the data elements. With big size of data elements, hash table is more efficiency than vector even though both run in 0(1) times.

## Concrete, Real-world Example

{Provide a concrete, real-world example application that uses operation(s) being analyzed.}

* In real – world, a hash table is using for data stored in databases of the key-value format such as google when we want to search for a key or for a subject that we want to know.
* In real -world

## Data Structure Selection

{Select the container best suited for the concrete, real-world example above. Your selection should be supported by the data collected and graphed. Explain why you selected the one you did, and why you did not select the others.}

# Removing from a SLL at the back versus a Vector at the front

## Data Analysis Graphs

|  |  |
| --- | --- |
| { insert your graph as a picture here} | { insert your graph as a picture here} |
| Figure ‑: {Give your graph a title here} | Figure ‑: {Give your graph a title here} |
| { insert your graph as a picture here} | { insert your graph as a picture here} |
| Figure ‑: {Give your graph a title here} | Figure ‑: {Give your graph a title here} |
| { insert your graph as a picture here} | |
| Figure ‑: {Give your graph a title here} | |

## Analysis and Summation

{Explain the information in the graph in terms of the operation’s efficiency class (Big-Oh). Talk to what you see on the graphs and explain why you see it. If the graphs are not what you expected, point that out and explain. Compare and contrast the operations performed on the different data structures. Identify and describe patterns in the graphs, including peaks, trends, and variability.}

{Do not indent or place blank lines between paragraphs.}

## Concrete, Real-world Example

{Provide a concrete, real-world example application that uses operation(s) being analyzed.}

## Data Structure Selection

{Select the container best suited for the concrete, real-world example above. Your selection should be supported by the data collected and graphed. Explain why you selected the one you did, and why you did not select the others.}

# Searching a Vector, DLL, SLL, BST, and a Hash Table

## Data Analysis Graphs

|  |  |
| --- | --- |
| { insert your graph as a picture here} | { insert your graph as a picture here} |
| Figure ‑: {Give your graph a title here} | Figure ‑: {Give your graph a title here} |
| { insert your graph as a picture here} | { insert your graph as a picture here} |
| Figure ‑: {Give your graph a title here} | Figure ‑: {Give your graph a title here} |
| { insert your graph as a picture here} | |
| Figure ‑: {Give your graph a title here} | |

## Analysis and Summation

{Explain the information in the graph in terms of the operation’s efficiency class (Big-Oh). Talk to what you see on the graphs and explain why you see it. If the graphs are not what you expected, point that out and explain. Compare and contrast the operations performed on the different data structures. Identify and describe patterns in the graphs, including peaks, trends, and variability.}

{Do not indent or place blank lines between paragraphs.}

## Concrete, Real-world Example

{Provide a concrete, real-world example application that uses operation(s) being analyzed.}

## Data Structure Selection

{Select the container best suited for the concrete, real-world example above. Your selection should be supported by the data collected and graphed. Explain why you selected the one you did, and why you did not select the others.}

# {You select #1}

## Data Analysis Graphs

|  |  |
| --- | --- |
| { insert your graph as a picture here} | { insert your graph as a picture here} |
| Figure ‑: {Give your graph a title here} | Figure ‑: {Give your graph a title here} |
| { insert your graph as a picture here} | { insert your graph as a picture here} |
| Figure ‑: {Give your graph a title here} | Figure ‑: {Give your graph a title here} |
| { insert your graph as a picture here} | |
| Figure ‑: {Give your graph a title here} | |

## Analysis and Summation

{Explain the information in the graph in terms of the operation’s efficiency class (Big-Oh). Talk to what you see on the graphs and explain why you see it. If the graphs are not what you expected, point that out and explain. Compare and contrast the operations performed on the different data structures. Identify and describe patterns in the graphs, including peaks, trends, and variability.}

{Do not indent or place blank lines between paragraphs.}

## Concrete, Real-world Example

{Provide a concrete, real-world example application that uses operation(s) being analyzed.}

## Data Structure Selection

{Select the container best suited for the concrete, real-world example above. Your selection should be supported by the data collected and graphed. Explain why you selected the one you did, and why you did not select the others.}

# {You select #2}

## Data Analysis Graphs

|  |  |
| --- | --- |
| { insert your graph as a picture here} | { insert your graph as a picture here} |
| Figure ‑: {Give your graph a title here} | Figure ‑: {Give your graph a title here} |
| { insert your graph as a picture here} | { insert your graph as a picture here} |
| Figure ‑: {Give your graph a title here} | Figure ‑: {Give your graph a title here} |
| { insert your graph as a picture here} | |
| Figure ‑: {Give your graph a title here} | |

## Analysis and Summation

{Explain the information in the graph in terms of the operation’s efficiency class (Big-Oh). Talk to what you see on the graphs and explain why you see it. If the graphs are not what you expected, point that out and explain. Compare and contrast the operations performed on the different data structures. Identify and describe patterns in the graphs, including peaks, trends, and variability.}

{Do not indent or place blank lines between paragraphs.}

## Concrete, Real-world Example

{Provide a concrete, real-world example application that uses operation(s) being analyzed.}

## Data Structure Selection

{Select the container best suited for the concrete, real-world example above. Your selection should be supported by the data collected and graphed. Explain why you selected the one you did, and why you did not select the others.}