CAB301 ASSIGNMENT

Patrick Cummins n10627138

MAY 24, 2021

Table of Contents

Introduction

This project is a software management system for a tool library. The system manages the tools as well as the members of the library. It allows for staff to organise, add, remove both tools and members. It allows members to hire and return tools as well as other quality of life functionality. Chief amongst the features is the ability for a member to view the top three most borrowed tools, which is implemented using a *Heap Sort* algorithm.

The purpose of this report is to demonstrate the design and complexity of the implemented Heap Sort algorithm, explaining its implementation as well as its efficiency. Further, the report will outline the tests plans and results for the specified functionality, demonstrating that the implementation is correct.

The report will give the pseudo code of the HeapSort algorithm, present an analysis of the algorithm, showings its time complexity and efficiency.
The report will then demonstrate the testing plans and results.

Algorithm Analysis

Pseudocode

Heap Bottom-Up Pseudocode

```
HeapBottomUp(H[1…n])
// Constructs a heap from elements of a given array
// by the bottom-up algorithm
// Input: An array H[1…n] of orderable items
// Output: A heap H[1…n]
for i <- [n / 2] down to 1 do
        k <- i; v <- H[k]
        heap <- false
        while not heap and 2 * k ≤ n do
        j <- 2* k
        if j < n // there are two children
                if H[j] < H[j + 1] j <- j + 1
        if v ≥ H[j]
                heap <- true
        else H[k] <- H[j]; k <- j
    H[k] <- v
```

Heapsort Pseudocode

```
HeapSort(A[0…n-1])
B[0…n-1] <- Ø
// Sorts array A into nondecreasing order
consider A as a complete binary tree and convert it into a heap using the HeapBottomUp procedure
// Runs the MaxKeyDelete three times to find the three biggest values, inputs them into an empty array
for v ← 0 to 3 do
        Use the MaximumKeyDeletion procedure to delete the root of the heap
        B[0…n-1] <- v

return B[0…n-1]
```

Max Key Delete Pseudocode

```
MaxKeyDelete(A[0…n-1], size)
// Exchange the root's key with the last key k of the heap
temp <- A[0]
A[0] <- A[size – 1]
A[size – 1] <- temp
// Decrease the heap's size by 1
```

```
n <- size – 1
// Heapify the complete binary tree
heap <- false
k <- 0
v <- A[0]
while not heap and 2 * k + 1 <- n – 1
    j <- 2 * k + 1 // left child of k
    if j < n – 1 // key has two children
        if A[j] < A[j + 1]
        j <- j + 1 // j is the larger child of k
    if v ≥ A[j]
        heap <- true
    else
        A[k] <- A[j]
        K <- j
A[k] <- v
return A[n]
```

Theoretical Algorithm Efficiency

Heap Bottom-Up

```
for i <- [n / 2] down to 1 do
        k <- i  O(1)
        v <- H[k] O(1)
        heap <- false O(1)
        while not heap and 2 * k ≤ n do
                j <- 2* k O(1)
                if j < n // there are two children
                        if H[j] < H[j + 1] O(1)
                                j <- j + 1 O(1)
                if v ≥ H[j]
                        heap <- true O(1)
                else
                        H[k] <- H[j] O(1)
                        k <- j O(1)
      H[k] <- v O(1)
```

O(log n)

Time Complexity = O(n log n)

Max Key Delete

```
temp <- A[0] O(1)
A[0] <- A[size – 1] O(1)
A[size – 1] <- temp O(1)
// Decrease the heap's size by 1
n <- size – 1 O(1)
// Heapify the complete binary tree
heap <- false O(1)
k <- 0 O(1)
v <- A[0] O(1)
while not heap and 2 * k + 1 <- n – 1
        j <- 2 * k + 1 // left child of k O(1)
        if j < n – 1 // key has two children
                if A[j] < A[j + 1] O(1)
                j <- j + 1  // j is the larger child of k O(1)
        if v ≥ A[j]
                heap <- true O(1)
        else
                A[k] <- A[j] O(1)
                K <- j O(1)
A[k] <- v O(1)
return A[n] O(1)
```

O(log n)

Time Complexity = O(log n)

Heap Sort
B[0…n-1] <- Ø O(1)
HeapBottomUp() O(n log n)
for v ← 0 **to** 3 **do**
Use the MaximumKeyDeletion procedure to delete the root of the heap O(log n)
    B[0…n-1] <- v O(1)

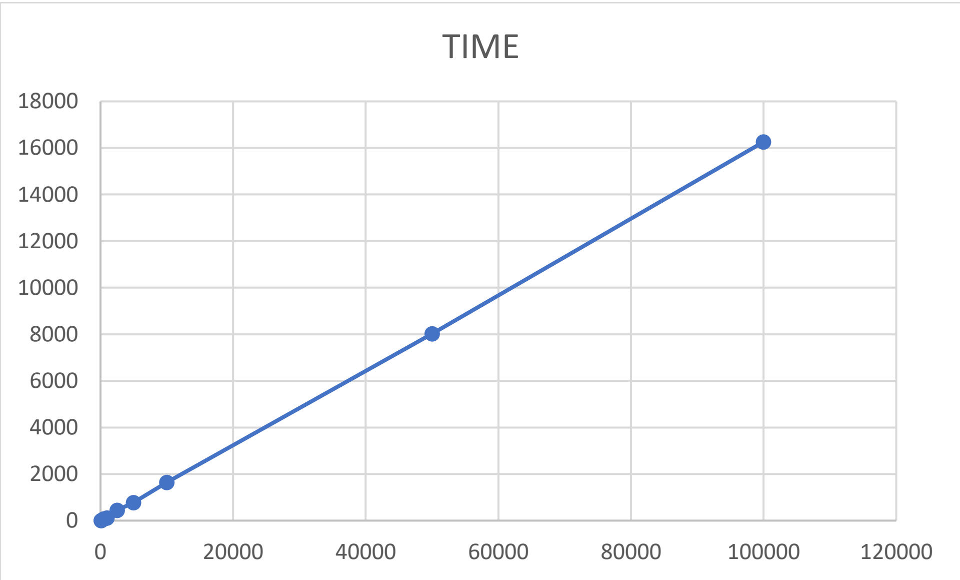
} O(n log n)

---

return *B*[0…n-1] O(1)
Time Complexity = O(n log n)

Empirical Algorithm Efficiency

To correctly test the algorithm empirically, multiple samples of arrays were created, loaded with random values. The program was setup to run the algorithm within a timer which would relay the time taken to complete the algorithm. The observed data was recorded and charted into a scatter plot seen below.



ARRAY SIZE	TIME m/s
100	10
500	71
1000	119
5000	778
10000	1641
50000	8021
100000	16260

An analysis of the data is as follows, taking two instances of *t*(2n) over *t*(n), their ratio was calculated to produce the efficiency of the algorithm.

$$\frac{t(10000)}{t(5000)} = \frac{1641}{778} = 2.10 \qquad \frac{t(100000)}{t(50000)} = \frac{16260}{8021} = 2.02$$

$$\frac{t(2n)}{t(n)} = \frac{(2n)\log(n)}{n\log n} = \frac{2\log(2n)}{\log n}$$

The mathematical results taken from the empirical tests show that the ratio of *t*(2n) and *t*(n) is greater than two, making it **less efficient than O(n)**.

The ratio is less than four, making it **more efficient than O(n^2).**

Therefore the efficiency class of the algorithm is O(n log n).

Software Tests

Staff Login

```
Please enter staff username...
staff_
```

```
Welcome to the Tool Library
===============Staff Menu================
1. Add a new tool
2. Add new pieces of an existing tool
3. Remove some pieces of a tool
4. Register a new member
5. Remove a member
6. Find the contact number of a member
0. Return to the main menu
=========================================
```

Add a new tool

```
Tool Library System - Add New Tool To Library
=============================================
Enter the name of the new Tool (0 to exit):
drill
Enter the quantity you wish to add:
5

Select category tool belongs to
1: Gardening Tools
2: Flooring Tools
3: Fencing Tools
4: Measuring Tools
5: Cleaning Tools
6: Painting Tools
7: Electronic Tools
8: Electricity Tools
9: Automotive Tools
Select option from menu (0 to exit):
1

Select the Tool Type
===================
1: Line Trimmers
2: Lawn Mowers
3: Hand Tools
4: Wheelbarrows
5: Garden Power Tools
Select option from menu (0 to exit):
1

drill has been added to the library, press any key to return to staff menu
```

Add pieces to a tool

```
Line Trimmers
==========================
0: Tool Name: tiny trimmer
Available Quantity: 1
===
1: Tool Name: small trimmer
Available Quantity: 1
===
2: Tool Name: medium trimmer
Available Quantity: 1
===
3: Tool Name: big trimmer
Available Quantity: 1
===
4: Tool Name: huge trimmer
Available Quantity: 1
===
5: Tool Name: drill
Available Quantity: 5
===
Select the number of the tool you wish to add pieces to...
0
How many pieces would you like to add?
1
Successfully added 1 pieces to the tool
Press 0 to return to staff menu
```

Remove pieces from a tool

```
Line Trimmers
==========================
0: Tool Name: tiny trimmer
Available Quantity: 2
===
1: Tool Name: small trimmer
Available Quantity: 1
===
2: Tool Name: medium trimmer
Available Quantity: 1
===
3: Tool Name: big trimmer
Available Quantity: 1
===
4: Tool Name: huge trimmer
Available Quantity: 1
===
5: Tool Name: drill
Available Quantity: 5
===
Select the number of the tool you wish to remove pieces from...
0
How many pieces would you like to remove?
1
Successfully removed 1 pieces from the tool
Press 0 to return to staff menu
```

Add a new member

```
Add a new member
================
Please enter a first name
andrew
Please enter a last name
sagorski
Please enter a mobile number
0400000000
Please enter a four digit pin
1234
You have successfuly added: andrew sagorski to the system
Press any key to return to staff menu
```

Remove a member

```
Delete a member
===============
Please enter the first name of the member you wish to delete
pat
Please enter the last name of the member you wish to delete
pat
Please enter the phone number of the member you wish to delete
0400000000
Please enter the pin of the member you wish to delete
1234
You have successfuly deleted: pat pat from the system
Press any key to return to staff menu
```

Find a member's number

```
Find a member's contact number
===============
Please enter the first name of the member
pat
Please enter the last name of the member
pat
pat pat's number: 0400000000
Press any key to return to staff menu
```

Member Login

```
=== Member Login ===
Please enter first name
pat
Please enter last name
pat
Please enter PIN
1234
```

```
Welcome to the Tool Library
================Member Menu================
1. Display all the tools of a tool type
2. Borrow a tool
3. Return a tool
4. List all the tools that I am renting
5. Display top three (3) most frequently rented tools
0. Return to the main menu
==========================================
```

Borrow tool

```
=== Borrow a Tool ===
=====================
Please enter the name of the tool you wish to borrow
tiny trimmer
Found tool Tool Name: tiny trimmer
Available Quantity: 1
You have successfully borrowed tiny trimmer
Press any key to return to member menu
```

Display tools by category/type

```
Display of Tools by Type
========================
1: Gardening Tools
2: Flooring Tools
3: Fencing Tools
4: Measuring Tools
5: Cleaning Tools
6: Painting Tools
7: Electronic Tools
8: Electricity Tools
9: Automotive Tools
Select option from menu (0 to exit):
```

```
Line Trimmers
==========================
Tool Name: tiny trimmer
Available Quantity: 1
===
Tool Name: small trimmer
Available Quantity: 1
===
Tool Name: medium trimmer
Available Quantity: 1
===
Tool Name: big trimmer
Available Quantity: 1
===
Tool Name: huge trimmer
Available Quantity: 1
===
Pess any key to return to member menu
```

Return Tool

```
=== Return a Tool ===
=====================
Please enter the name of the tool you wish to return
tiny trimmer
Found tool Tool Name: tiny trimmer
Available Quantity: 0
Found tool Tool Name: tiny trimmer
Available Quantity: 1
You have successfully returned:
Tool Name: tiny trimmer
Available Quantity: 1

Please press any key to return to member menu
```

Display borrowed Tools

```
=== Displaying Borrowed Tools ===
=================================
Tool Name: tiny trimmer
Available Quantity: 0
===
Press any key to return to member menu
```

Top 3 Borrowed Tools

```
=== Top Three Most Borrowed Tools ===
=====================================
#1: huge trimmer
Times Borrowed: 4
===
#2: big trimmer
Times Borrowed: 3
===
#3: medium trimmer
Times Borrowed: 2
===
```