**A picture containing text

Description automatically generated**

**School Of Information Technology**

**IT2553 DS & Algo**

|  |  |
| --- | --- |
| **Admin No & Team Members Name:** | 201520M: Eden Will Sng Jin Xuan |
| **PEM Group:** | SF2102 |
| **Module:** | IT2556 |
| **Assignment:** | Tutorial 03 |

1. Unsorted list takes 7 comparisons
2. Improved Version takes 6 sorts

(Sorted list requires 10 comparisons (sequential searches through whole list) )(\*improved version compares the list until it compares a number which isn’t able to compare further anymore) (improved version takes 6 comparisons or fewer item is not found)

1. Given the following list of numbers and their corresponding index position

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index Position | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| List Element | 10 | 23 | 25 | 34 | 36 | 42 | 63 | 74 | 87 | 92 | 99 |

Binary search to search value 63

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Low Pointer Index Position | Middle Pointer Index Position | High Pointer Index Position | Found(Yes/No) | Value |
| 1 | 0 | 5 | 10 | No | 42 |
| 2 | 6 | 8 | 10 | No | 87 |
| 3 | 6 | 6 | 7 | yes | 63 |

4.  
Binary search to search value 18

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Low Pointer Index Position | Middle Pointer Index Position | High Pointer Index Position | Found(Yes/No) | Value |
| 1 | 0 | 5 | 10 | No | 42 |
| 2 | 0 | 2 | 4 | No | 25 |
| 3 | 0 | 1 | 1 | No | 23 |
| 4 | 1 | 1 | 1 | No | 23 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Pass | Low Pointer Index Position | Middle Pointer Index Position | High Pointer Index Position | Found(Yes/No) | Value |
| 1 | 0 | 5 | 10 | No | 42 |
| 2 | 0 | 2 | 4 | No | 25 |
| 3 | 0 | 0 | 1 | No | 23 |
| 4 | 1 | 1 | 1 | No | 23 |
| 5 | 1 | - | 0 | No  Low ->  High Pointer -> value 18 is not in the list   while low<=high :  Rmb this | - |

Return -1

Midpoint = Low + High / 2 (round down)