Department of Information Systems and Technologies

CTIS 152 – Data Structures and Algorithms

Spring 2024 - 2025

Lab Guide #11 – Week 6-2

OBJECTIVE : Binary Search

Instructors : Serpil TIN
Assistants : Berk ÖNDER &Hatice Zehra YILMAZ

BINARY SEARCH ALGORITHM:

```
1. Let top be the subscript of the initial array element.
2. Let bottom be the subscript of the last array element.
3. Repeat until top exceeds bottom, thus there are no more elements to check
   4. Let middle be the subscript of the element halfway from top to bottom.
    5. If the element at middle is the target, than return middle.
       6. else  if the element at middle is larger than the target, let bottom be middle-1,
        thus continue the search in the first half.
   7. else let top be middle+1, thus continue the search in the second half.
8. Return -1 since the loop terminated, but the number is not found.
```

Q1. Write a C program that reads the employee IDs of a department from an **ascendingly** sorted file named **"empIDs.txt"** into an integer array with a **maximum size of 20**. Then, it gets a number from the user to search in the text file and displays the result as shown in the example run. Use a **Binary Search Algorithm** and display how many comparisons were made to find the value.

Write the following functions;

- **read:** that takes the file pointer, and ID array as input parameters. Then, it reads the IDs from the file into the array (MAX array size is 20) and returns the actual number of elements in the array.
- **binarySearch:** that takes the ID array, the number of elements in the array, and an ID to search as parameters. Then searches the ID in the array using the Binary Search Algorithm, if found returns the index as middle, otherwise returns -1. The function should also return the number of search steps. **You may find the Binary Search Algorithm above ;)**

Example Run #1 :

```
Enter value to find : 1694
1694 is found at 5. line
4 comparisons.
```

Example Run #2 :

```
Enter value to find : 1111
The number is NOT found in the empIDS.txt
4 comparisons.
```

Example Run #3 :

```
Enter value to find : 2113
2113 is found at 12. line
2 comparisons.
```

empIDs.txt

```
1254
1369
1487
1584
1694
1788
1894
1952
2001
2010
2056
2113
2269
2358
2471
```

Project Name: LG11_Q1
File Name: Q1.cpp

Q2. Write a C program that reads the software company names from a **descendingly** sorted file named **"companies.txt"** file into a string array with a **maximum size of 30**. Then, the program reads a company name to search from the user and searches it through the company list by using the **Binary Search Algorithm**. If the searched company name is found, an appropriate message will be displayed as in the example run. The program stops searching for a company name when the user enters "**end**" as a company name.

Write the following functions: **`readFromFile, display, binarySearch`**

`companies.txt`

```
WookWeb Creative Agency
WesterOps
Wag The Dog
Vis Marin Bilisim
VevaSoft
Varien Software
Terra Software
Pomelo Soft
Most Idea
Helezon Software
FeriSoft
Code Section
BWA Digital
BKB Software
AR AITech
```

`Example Run:`

```
The List of Companies
-------------------
WookWeb Creative Agency
WesterOps
Wag The Dog
Vis Marin Bilisim
VevaSoft
Varien Software
Terra Software
Pomelo Soft
Most Idea
Helezon Software
FeriSoft
Code Section
BWA Digital
BKB Software
AR AITech

Enter a company name to search (or end to stop searching): VevaSoft
VevaSoft found on the index 4 in the list

Enter a company name to search (or end to stop searching): AR AITech
AR AITech found on the index 14 in the list

Enter a company name to search (or end to stop searching): AkinSoft
AkinSoft could not be found!

Enter a company name to search (or end to stop searching): end
```

Project Name: LG11_Q2
File Name: Q2.cpp

Q3. Write a C program that reads the phone information from the file named **"phone.txt"** into a **dynamically created** structure array. The first row of the file contains the file size. Phone information includes; **phone names, prices, and stock quantities**. The program reads the number of phones to search from the user and searches it through the phone information by using the **binary search** algorithm. If the searched number is found, an appropriate message will be displayed as in the example run. The program stops searching for the number of people when the user enters "**-1**" as a number.

The file contains ascending order according to the stock quantities.

Write the following functions: **`readFromFile, display, binarySearch`** for structure array**.**

`phone.txt`

```
5
Google 1000 8
Apple 1200 10
OnePlus 800 12
Samsung 900 15
Xiaomi 700 20
```

`Example Run:`

```
     PHONE INFORMATIONS FOR PEOPLE
-----------------------------------------
Google              1000           8
Apple               1200          10
OnePlus              800          12
Samsung              900          15
Xiaomi               700          20

Enter number of stock to see the campaign: 8
Google  found on the 1000 sale rate 8 stock

Enter number of stock to see the campaign: 10
Apple  found on the 1200 sale rate 10 stock

Enter number of stock to see the campaign:15
Samsung  found on the 900 sale rate 15 stock

Enter number of stock to see the campaign: -1
```

Project Name: LG11_Q3
File Name: Q3.cpp

ADDITIONAL QUESTIONS

AQ1. Write a C program that forms a hotel list by getting hotel names from the text file "**hotels.txt**" into a string array with a **maximum size of 30**, sorts the list in **ascending order** by using the **Bubble Sort Algorithm,** and then displays the sorted list. The program displays the following menu and does the necessary operations according to the user's choice;

```
          MENU
********************
1. Display the Hotel List
2. Add a new Hotel
3. EXIT
```

When the user selects the "Add a new Hotel" option; the program asks for the new hotel name and checks the hotel if it is already on the list or not, displays a warning message, or adds the new hotel to the **correct position** in the list. The program stops when the user selects the EXIT option from the menu.

Write the following functions;

- **menu:** displays a menu, read-validates, and returns the user's choice.
- **readFromFile:** reads the hotel names from the file into a string array and returns the actual size of the array.
- **display:** displays the hotel list.
- **bubbleSort:** sorts the hotel list in ascending order.
- **binarySearch:** searches for the specified hotel name in the hotel list and returns the index of the searched hotel. If the hotel is NOT in the list, the function returns **-1** and the **correct position** to insert the hotel name.
- **shiftDown:** shifts down the hotels' names to open a space to the given position.
- **addToList:** adds the new hotel to the hotel list using the function **shiftDown**.

hotels.txt

```
Hilton Istanbul
Merit Crystal
JW Marriott Miami
Wow Cremlin Palace
Cambria Beach Lodge
Merit Royal
Hilton SAii Lagoon Maldives
```

Project Name: LG11_AQ1
File Name: AQ1.cpp

Example Run:

```
Hotel List
--------------------
1) Cambria Beach Lodge
2) Hilton Istanbul
3) Hilton SAii Lagoon Maldives
4) JW Marriott Miami
5) Merit Crystal
6) Merit Royal
7) Wow Cremlin Palace

MENU
********************
1. Display the Hotel List
2. Add a new Hotel
3. EXIT

Enter your choice: 2

Enter the hotel name: Hilton Istanbul
"Hilton Istanbul" already exists in the list!


MENU
********************
1. Display the Hotel List
2. Add a new Hotel
3. EXIT

Enter your choice: 2

Enter the hotel name: Konya Dedeman
"Konya Dedeman" ADDED to the list!
```

```
MENU
********************
1. Display the Hotel List
2. Add a new Hotel
3. EXIT

Enter your choice: 1

Hotel List
--------------------
1) Cambria Beach Lodge
2) Hilton Istanbul
3) Hilton SAii Lagoon Maldives
4) JW Marriott Miami
5) Konya Dedeman
6) Merit Crystal
7) Merit Royal
8) Wow Cremlin Palace

MENU
********************
1. Display the Hotel List
2. Add a new Hotel
3. EXIT

Enter your choice: 6

MENU
********************
1. Display the Hotel List
2. Add a new Hotel
3. EXIT

Enter your choice: 3
```

AQ2. Write a C program that gets phone information (**phone ID, price, brand, and model**) from the file named **"phones.txt"** into a structure array with a maximum size of **20**. The program displays a menu with the following options, does the necessary operations depending on the user's choice, and stops when the user selects the EXIT option from the menu:

```
              MENU
*************************************
1. Display Phone List
2. Search for a brand
3. Sort by price (Ascending/Descending) Order
4. EXIT
```

The Program does the following operations;

- displays the phone list on the screen,
- reads a brand to search from the user and displays all the phones with the given brand on the screen. Also, the number of models should be displayed. If there is no model with the specified brand in the store, a warning message will be displayed.
- depending on the user's choice (1: Ascending/2: Descending), the program sorts the data according to phone **prices** using the bubble sort algorithm. Then the displays the sorted array.

Write the following functions: **menu, readFromFile, display, bubbleSort, searchBrand**

Phones.txt

```
1234 4753.00 Huawei P40 Lite 128 GB
1452 7139.50 Xiaomi Mi 11 Lite 128 GB
2145 5164.20 Huawei Nova 8i 128 GB
2424 10535.00 Xiaomi Mi 11T 256 GB
2525 5466.67 Samsung Galaxy A51 2020 128 GB
3213 4898.99 Oppo Reno 5 Lite 128 GB
3347 18998.00 iPhone 13 128 GB
3538 13158.76 Samsung Galaxy Z Flip3 5G 128 GB
4343 13899.00 Apple iPhone 11 128 GB
4444 17184.00 Samsung Galaxy S21 Ultra 5G 128 GB
```

Project Name: LG11_AQ2
File Name: AQ2.cpp

Example Run:

```
      MENU
*************************************
1. Display Phone List
2. Search for a brand
3. Sort by price (Ascending/Descending) Order
4. EXIT

Please Enter your choice: 6

Please Enter your choice: 1

 Id   Brand      Model                       Price
--------------------------------------------------------
 1234 Huawei     P40 Lite 128 GB              4753.00 TL
 1452 Xiaomi     Mi 11 Lite 128 GB            7139.50 TL
 2145 Huawei     Nova 8i 128 GB               5164.20 TL
 2424 Xiaomi     Mi 11T 256 GB               10535.00 TL
 2525 Samsung    Galaxy A51 2020 128 GB       5466.67 TL
 3213 Oppo       Reno 5 Lite 128 GB           4898.99 TL
 3347 iPhone     13 128 GB                   18998.00 TL
 3538 Samsung    Galaxy Z Flip3 5G 128 GB    13158.76 TL
 4343 Apple      iPhone 11 128 GB            13899.00 TL
 4444 Samsung    Galaxy S21 Ultra 5G 128 GB  17184.00 TL

   MENU
*************************************
1. Display Phone List
2. Search for a brand
3. Sort by price (Ascending/Descending) Order
4. EXIT

Please Enter your choice: 2

Enter a brand to search: Samsung

 2525 Samsung    Galaxy A51 2020 128 GB       5466.67 TL
 3538 Samsung    Galaxy Z Flip3 5G 128 GB    13158.76 TL
 4444 Samsung    Galaxy S21 Ultra 5G 128 GB  17184.00 TL
There are 3 models of Samsung phones
```

```
   MENU
*************************************
1. Display Phone List
2. Search for a brand
3. Sort by price (Ascending/Descending) Order
4. EXIT

Please Enter your choice: 3

Please select your sort decision (1: Ascending, 2: Descending): 1

 Id   Brand      Model                       Price
--------------------------------------------------------
 1234 Huawei     P40 Lite 128 GB              4753.00 TL
 3213 Oppo       Reno 5 Lite 128 GB           4898.99 TL
 2145 Huawei     Nova 8i 128 GB               5164.20 TL
 2525 Samsung    Galaxy A51 2020 128 GB       5466.67 TL
 1452 Xiaomi     Mi 11 Lite 128 GB            7139.50 TL
 2424 Xiaomi     Mi 11T 256 GB               10535.00 TL
 3538 Samsung    Galaxy Z Flip3 5G 128 GB    13158.76 TL
 4343 Apple      iPhone 11 128 GB            13899.00 TL
 4444 Samsung    Galaxy S21 Ultra 5G 128 GB  17184.00 TL
 3347 iPhone     13 128 GB                   18998.00 TL

   MENU
*************************************
1. Display Phone List
2. Search for a brand
3. Sort by price (Ascending/Descending) Order
4. EXIT

 Please Enter your choice: 4
```