NAME : FARICHA AULIA
NIM : 2141720155
ABSENT : 08
CLASS : 1I - IT
TOPIC : JOBSHEET

6.1. Purpose

After doing this lab exercise, students are able to:

1. Explain about the Searching algorithm.
2. Create and declare the structure of the Searching algorithm
3. Implement and implement the Searching algorithm.

6.2. Searching / Search Using Sequential Search Algorithm

Pay attention to the Student class diagram below! This class diagram will then be made as a reference in making the Student class program code.

Based on the class diagram above, a Student class will be created which is used to create student objects that will be stored in an array. There is a parameterized constructor and a dispStudent() method to display all the existing attributes

Furthermore, the class diagram above is a representation of a class that is used to perform operations on the student array object, for example to add a student object, display all student data, to perform a search based on NIM using the Sequential Search algorithm, displaying position search results (index), as well as displaying student data search results.

6.2.1. Lab unit 1 : Sequential Search

1. Create a new project with the name Searching
2. Create a new project with the name week7.
3. Create Student00 class, replace 00 with your absent number. then declare the following attributes
4. Create parameterize constructor Student00 with parameter (int n, String na, int a, double g) then Assign the constructor with the following code
5. Create dispStudent() method with void type
6. Create new class with the name StudentSearch00, replace 00 with your absent number!
7. Create addStudent() method inside StudentSearch00 class! This addStudent() method used to add an object of class Student to the StudentList attribute.
8. create display() method inside StudentSearch00 class! This Method display() method is used to display all the data of students in the class! Note that the foreach syntax is somewhat different from the previously studied for, although it is conceptually similar
9. create FindSeqSearch() method with int type and the add paramater key with integer datatype. Then create the contents of the FindSeqSearch method with a search algorithm using the sequential search technique.
10. Create DispPosition method like the following code!. This method is used to display the index search results.
11. Create DispData method like the following code!. This method is used to display the student data search results.
12. Create StudentMain00 class. Replace 00 with your absent number! add the main method as shown in the following image!
13. Inside main()method, create an object of class StudentSearch and create 5 student objects then add all student objects by calling the addStudent() method.
14. Inside main()method, Call display() method to see all the data.
15. To do a search based on the student's NIM. Create key variable that can accommodate input from the keyboard and then call the FindSeqSearch() method with the key as a parameter.

NAME : FARICHA AULIA
NIM : 2141720155
ABSENT : 08
CLASS : 1I - IT
TOPIC : JOBSHEET

16. Call DispPosition() method from StudentSearch class
17. Call DispData()method from StudentSearch class.
18. Run StudentMain Class.

6.2.3. Question

1. Explain the difference between DispPosition()and DispData() methods in the StudentSearch class

Answer:

- dispData to enter student data
- dispPosition to display the data found at a certain index

2. What is the use of break keyword in the program code below!Explain!

Answer: to stop the loop that is executed according to the condition

3. If the input data Nim entered is not sorted from small to large. Can the program still run? Are the results correct? Explain Why!

Answer: you can, because the program is already filled in to sort the data

6.3. Searching Using Binary Search Algorithm

6.3.1. Lab Unit 2: Binary Search

1. Create FindBinarySearch() method with integer type inside StudentSearch class. Then create the contents of the FindBinarySearch method with a search algorithm using the Binary search technique.
2. Inside StudentMain Class, Call FindBinarySearch method. Then call DispPosition()and DispData()method.
3. Run StudentMain Class.

6.3.3. Question

1. Show the program code where the divide process is executed!

Answer:
```
if(right>=left){

        mid=(left+right)/2;
```

2. Show the program code where the conquer process is executed!

Answer:
```
if(key==StudentList[mid].nim){

            return(mid);
        }
        else if(StudentList[mid].nim>key){
            return FindBinarySearch(key, left, mid-1);
        }else{
            return FindBinarySearch(key, mid+1, right);
        }
```

3. If the input data Nim entered is not sorted. Can the program still run? Are the results correct? Explain Why!

Answer: you can, because the program is already filled in to sort the data

4. If the input Nim is entered from the largest to the smallest NIM (Descending order) (for example : 20215, 20214, 20212, 20211, 20210) and the element you are looking for is 202110. What are the results of a binary search? Are the results correct? If it doesn't match, then modify the binary search program code so that the results are correct!

NAME : FARICHA AULIA
NIM : 2141720155
ABSENT : 08
CLASS : 1I - IT
TOPIC : JOBSHEET

Answer: 5 binary

6.4. Divide and Conquer Enrichment Experiment

6.4.1. Lab-unit 3 : Merge Sort

1. Inside StudentSearch00 class, create mergeSort() method.
2. Inside StudentSearch00 class, create merge() method to perform the process of merging data from the left and right.
3. Inside StudentSearch00 class, create sort() method with tree parameters : data, left index, and right index
4. The final step, in the StudentMain class modifiy the program. change statement that you can input undordered student data and call mergeSort() method before displaying ALL student data.

src > week7 > StudentMain08.java > ...

```java
package week7;
import java.util.Scanner;

public class StudentMain08 {
    Run | Debug
    public static void main(String[]args){
        Scanner scInt = new Scanner(System.in);
        Scanner scStr = new Scanner(System.in);

        StudentSearch08 data = new StudentSearch08();
        int Std = 5;
        System.out.println("==================================");
        System.out.println("Input Student Data Ascending Order based on NIM");
        for(int i=0;i<Std;i++){
            System.out.println("==================================");
            System.out.println("NIM\t : ");
            int nim=scInt.nextInt();
            System.out.println("Name\t : ");
            int name=scInt.nextInt();
            System.out.println("Age\t : ");
            int age=scInt.nextInt();
            System.out.println("GPA\t : ");
            int gpa=scInt.nextInt();

            Student08 s = new Student08(nim, name, age, gpa);
            data.addStudent(s);
        }

    System.out.println("==================================");
    System.out.println("All Student Data");
    data.display();

    System.out.println("==================================");
    System.out.println("Search data");
    System.out.println("Input NIM you are looking for");
    System.out.print("NIM");
    int key = scInt.nextInt();

    System.out.print("Using sequential search algorithm");
    int position = data.FindSeqSearch(key);

    System.out.print("Using binary search algorithm");
    position = data.FindBinarySearch(key, 0, Std-1);
    data.DispPosition(key, position);
    data.DispData(key, position);
    }
}
```

NAME : FARICHA AULIA
NIM : 2141720155
ABSENT : 08
CLASS : 1I - IT
TOPIC : JOBSHEET

src > week7 > Student08.java > ...

```java
package week7;

public class Student08 {
    int nim;
    String name;
    int age;
    double GPA;

    Student08(int n, String na, int a, double g){
        nim=n;
        name=na;
        age=a;
        GPA=g;
    }

    void dispStudent(){
        System.out.println("NIM  : " +nim);
        System.out.println("Name : " +name);
        System.out.println("Age  : " +age);
        System.out.println("GPA  : " +GPA);
    }
}
```

src > week7 > StudentSearch08.java > ...

```java
package week7;

public class StudentSearch08 {
    Student08 StudentList[]=new Student[5];
    int index;

    void addStudent(Student08 s){
        if(index<StudentList.length){
            StudentList[index]=s;
            index++;
        }else{
            System.out.println("Data is Full");
        }
    }

    void display(){
        for(Student08 s : StudentList){
            s.dispStudent();
            System.out.println("======================");
        }
    }

    public int FindSeqSearch(int key){
        int pos = -1;
        for(int j=0; j<StudentList.length; j++){
            if(StudentList[j].nim==key){
                pos=j;
                break;
            }
        }
        return pos;
    }

    void DispPosition(int x, int pos){
        if(pos!=-1){
            System.out.println(x+"data found at index : " +pos);
        }else{
            System.out.println(x+"data not found");
        }
    }
```

NAME : FARICHA AULIA
NIM : 2141720155
ABSENT : 08
CLASS : 1I - IT
TOPIC : JOBSHEET

```
    void DispData(int x, int pos){
        if(pos!=-1){
            System.out.println("Nim  : "+x);
            System.out.println("Nama : "+StudentList[pos].name);
            System.out.println("Age  : "+StudentList[pos].age);
            System.out.println("GPA  : "+StudentList[pos].GPA);
        }else{
            System.out.println(x+"data not found");
        }
    }

    public int FindBinarySearch(int key,int left, int right){
        int mid;
        if(right>=left){
            mid=(left+right)/2;
            if(key==StudentList[mid].nim){
                return(mid);
            }
            else if(StudentList[mid].nim>key){
                return FindBinarySearch(key, left, mid-1);
            }else{
                return FindBinarySearch(key, mid+1, right);
            }
        }
        return -1;
    }

    public void mergeSort(){
        sort(StudentList, 0, StudentList.length-1);
    }
```

```
    public void merge(Student08 data[], int left, int middle, int right){
        Student08[] temp = new Student08[data.length];
        for(int i=left; i<=right; i++){
            temp[i] = data[i];
        }
        int a = left;
        int b = middle+1;
        int c = left;

        while(a<=middle&&b<=right){
            if(temp[a].nim <= temp[b].nim){
                data[c] = temp[a];
                a++;
            }else{
                data[c] = temp[b];
                b++;
            }
            c++;
        }
        int s = middle - a;
        for(int i=0; i<=s; i++){
            data[c+i] = temp[a+i];
        }
    }

    public void sort(Student08 data[], int left, int right){
        if(left<right){
            int middle=(left+right)/2;
            sort(data, left, middle);
            sort(data, middle+1, right);
            merge(data, left, middle, right);
        }
    }

}
```

NAME : FARICHA AULIA
NIM : 2141720155
ABSENT : 08
CLASS : 1I - IT
TOPIC : JOBSHEET

6.5. Exercise

1. Modify the searching experiment above with the following conditions:
Before searching with binary search algorithm, sorting must be done using a sorting algorithm (choose one sorting algorithm from the previous meeting: bubble, selection, or insertion)

```
63      public void bubbleSort(){
64          for (int i=0; i<StudentList.length-1; i++){
65              for(int j=0; j<StudentList.length-i-1;j++){
66                  if(StudentList[j].GPA<StudentList[j+1].GPA){
67                      //Swapping Process
68                      Student08 tmp = StudentList[j];
69                      StudentList[j] = StudentList[j+1];
70                      StudentList[j+1]=tmp;
71                  }
72              }
73          }
74      }
```

2. Modify the searching experiment above with the following conditions: - Search is done by Student Name (use Sequential Search Algorithm) - If there is the same name? How does the output of the program code when a search is performed on the same name!

```
public int FindSeqSearchName(int key){
    int name = -1;
    for(int j=0; j<StudentList.length; j++){
        if(StudentList[j].name==key){
            name=j;
            break;
        }
    }
    return name;
}
```

3. There is a 1-dimensional data array as follows

0	1	2	3	4	5	6	7	8	9
12	17	2	1	70	50	90	17	2	90

Write a program to sort the array (you can choose the sorting method) then do a search and print the array with the largest value, and print how many values are there, and display it on all indexes of the largest value! (using binary search)

NAME	:	FARICHA AULIA
NIM	:	2141720155
ABSENT	:	08
CLASS	:	1I - IT
TOPIC	:	JOBSHEET

src > week7 > exercise3.java > ...

```java
package week7;

public class exercise3 {
    Run | Debug
    public static void main(String[] args) {
        int A[] = new int [10];
        A[0] = 11;
        A[1] = 3;
        A[2] = 4;
        A[3] = 2;
        A[4] = 6;
        A[5] = 8;
        A[6] = 7;
        A[7] = 9;
        A[8] = 10;
        A[9] = 5;
        int i,j,N = 10, temp;
        System.out.println("DISPLAY ARRAY OF THE SMALLEST");
        for (i = 0; i < N; i++) {
            for (j = N-1; j>i; j--){
                if (A[j] < A[j-1]){
                temp=A[j];
                A[j]=A[j-1];
                A[j-1]=temp;
                }
            }
        }
        for (j = 0; j < N; j++) {
            System.out.print(A[j]);
        }
        int max;
        max = A[0];
        for(i = 0; i < N; i++) {
            if (A[i] > max){
              max = A[i];
              temp = i+1;
            }
          }
      }
}
```