CEGEP VANIER COLLEGE CENTRE FOR CONTINUING EDUCATION Programming Algorithms and Patterns 420-930-VA

Teacher: Samir Chebbine Lab 1 Jul 14, 2023

Lab 1: Arrays, Arrays of objects and ArrayList

Complete all these following programs as explained during **Zoom Synchronous classes.** All *missing coding statements* were provided there with explanation. Create and Submit a Word file *Lab100PProgramminAlgorithmsYourName.docs* which includes output screenshots for every Java Project. Submit the Java projects too.

1. Data Structure

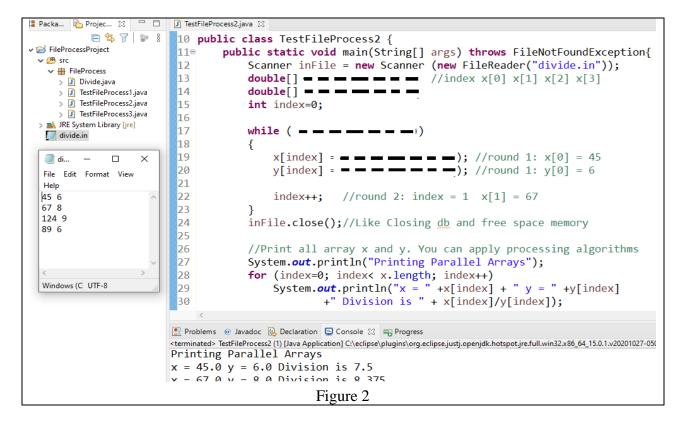
a) Reading Data From Text File without Array

Create a Java Project to be named *FileProcessProject* using Eclipse IDE for reading input file *divide.in*. Create *TestFileProcess1.java* to read from input file as shown in Figure 1.

```
🚦 Packa... 🔒 Projec... 💢 🖳 🗍 TestFileProcess1.java 💥
         □ 🕏 🎖 🐌 👂 public class TestFileProcess1 {
10
 🗸 进 src
                       11⊝
                               public static void main(String[] args) throws FileNotFoundException{
  // TODO Auto-generated method stub
                       12
    > Divide.java
                       13
    > TestFileProcess1.java
    > TestFileProcess2.java
                       14
                                   Scanner inFile = new Scanner (new FileReader("divide.in"));
    > I TestFileProcess3.java
                       15
                                   double x, y;
 JRE System Library [jre]
                        16
  divide.in
                        17
                                   while ( - - - - - -
                        18
                        19
                                       20
                                       y = - - - - - ); //round 1: y = 6
                        21
                        22
                                       System.out.println("x = " + x + " y = " + y
                        23
                                               +"Division is " + x/y);
                        24
                                   }
                       📳 Problems @ Javadoc 🔒 Declaration 📮 Console 🛭 🖏 Progress
                      <terminated> TestFileProcess1 (2) [Java Application] C:\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_15.0.1.v20201027-0507
                      x = 45.0 y = 6.0Division is 7.5
                      x = 67.0 y = 8.0Division is 8.375
                      x = 124.0 y = 9.0Division is 13.7777777777779
                      Figure 1
```

b) Reading Data From Text File and Storing it into Parallel Arrays

Create *TestFileProcess2.java* to read from input file and storing its content into parallel arrays as shown in Figure 2.



c) Reading Data From Text File and Storing it into Object of Arrays

Create *TestFileProcess3.java* to read from input file and storing its content into object of array as shown in Figure 3.

```
Projec... 🎖 🖳 🗋 🚺 TestFileProcess3.java 🛱
                         E 🕏 🥫 🕯 10 public class TestFileProcess3 {

→ BrileProcessProject

→ BrileProcessProcessProject

→ BrileProcessProject

→ BrilePro
                                                             11
   12⊝
                                                                                public static void main(String[] args) throws FileNotFoundException{
       // TODO Auto-generated method stub
                                                          213
           > Divide.java
            TestFileProcess1.java
TestFileProcess2.java
TestFileProcess3.java
                                                             14
                                                             15
                                                                                            Scanner inFile = new Scanner (new FileReader("divide.in"));
                                                                                           Divide[] arrayInFile = - - - -; //index arrayInfile[0] arrayInfile[1]
      ■ JRE System Library [jre]
                                                             16
                                                             17
      divide.in
                                                                                            int index=0:
                                                             18
                                                                                            for (index=0; index< arrayInFile.length; index++)</pre>
                                                             19
       divide - Notep...
                                                                                                      arrayInFile[ -
                                                              20
      File Edit Format 1
                                                             21
    456
678
                                                              22
                                                                                            index = 0;
                                                                                            while (inFile.hasNextLine())
                                                             23
      124 9
      89 6
                                                              24
                                                                                                                                                                        — — — — — — //round 1: arrayInFile[0].
                                                              25
                                                                                                       arrayInFile[index].set
                                                                                                    arrayInFile[index].set - - - - - //round 1: arrayInFile[0].
                                                              26
                                                              27
                                                              28
                                                                                                      index++; //round 2: index = 1 arrayInFile[1].x = 67
                                                              29
                                                              30
                                                                                            inFile.close();//Like Closing db and free space memory
                                                              31
                                                              32
                                                                                            //Print all array x and v
                                                              33
                                                                                            //you can apply processing algorithms
                                                              34
                                                                                            System.out.println("Printing Array of Object");
                                                              36
                                                                                           37
                                                              38
                                                              39
                                                              40
                                                              41
                                                                                                                            arrayInFile[index].getX()/arrayInFile[index].getY());
                                                           42
                                                             43

    Problems @ Javadoc    Declaration    □ Console    □ Progress

                                                          <terminated> TestFileProcess3 [Java Application] C\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_15.0.1.v20201027-0507\jre\bin\javaw.exe
PTINLING ATTAY OT UDJECL
                                                           x = 45.0 y = 6.0 Division is 7.5
                                                          x = 67.0 y = 8.0 Division is 8.375
                                                                                                                                                 Figure 3
```

2. ArrayLists

a) Create *ArrayListProcessProject* using Eclipse IDE for reading input file *divide.in*. Create *TestArrayList1.java* to to create ArrayList from List interface as shown in Figure 4.

```
Packa... Projec... ⊠ □ □ ② TestArrayList1java ⊠ □ ② ▼ □ □ package ArrayListProcess;
ArrayListProcessProject
 ₩ SEC
                            3 import java.util.*;
   > Divide java
                          5 public class TestArrayList1 {
6
7= public static void main(S
     >  TestArrayList1.java
     > I TestFileArrayList2.java
   A JRE System Library [jre]
                                    public static void main(String[] args) {
divide.in
FileProcessProject
                            9
                                         List<String> namelist = new ArrayList ==
LinkedList1Project
                           10
                                        String [ ] names = {"Ann", "Bob", "Caror };
                           11
                                        int index=0;
                           12
                                         //Reading from Array names and filling up namelist arrayList
                           14
                                        for (index=0; index< names.length; index++)
                                             namelist.add(names[index]);
                           15
                           16
                           17
                                         //Print ArrayList namelist
                                        for (index=0; index< namelist.size(); index++)</pre>
                           18
                                             System.out.println(
                           19
                           20
                                    }
                           21
                           22 }
                                                    Figure 4
```

b) Reading Data from Text File and Storing it into ArrayList Data Structure Create *TestFileArrayList2.java* to read from input file and storing its content into ArrayList as shown in Figure 5.

```
    ♣ Packa...
    ♠ Projec...
    ☒
    □
    ☑
    ☐ TestArrayList1.java
    ☑
    ☐ TestFileArrayList2.java
    ☒

    ♠
    ☒
    ☒
    ☒
    ■
    ■
    ■
    □
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■
    ■

                                            8⊜
9
                                                          public static void main(String[] args) throws FileNotFoundException{

✓ 

✓ ArrayListProcessProject

                                                                  Scanner inFile = new Scanner (new FileReader("divide.in"));
   10
                                                                  List - - inFileArrayList = new ArrayList - - ();
                                             11
         > Divide.iava
         >  TestArrayList1.java
>  TestFileArrayList2.java
                                             12
                                                                 int index=0:
                                                                  //Reading from Input File names and filling up inFileArrayList
                                             13
     ■ JRE System Library [jre]
                                                                   while (inFile.hasNextLine())
                                             14
     divide.in
                                             15
                                                                                                                                                      🥐 Problems @ Javadoc 📵 Declara
> 124 FileProcessProject
                                                                          Divide divideObj = new Divide();
                                             16
                                                                                                                                                      <terminated> TestFileArrayList2 [Java Ap
                                                                           divideObj.setX(inFile.nextDouble());
                                             17
                                                                                                                                                      x= 45.0 y= 6.0
                                             18
                                                                          divideObj.setY(inFile.nextDouble());
                                                                                                                                                      x = 67.0 y = 8.0
                                                                                                                                                      x = 124.0 y = 9.0
                                                                           inFileArrayList.add - - - -
                                                                                                                                                      x = 89.0 y = 6.0
                                             22
                                                                   inFile.close();//Like Closing db and free space memory
                                             23
                                             24
                                             25
                                                                   //Print ArrayList inFileArrayList
                                                                   26
                                             27
                                                                                               y = = = = .get(index).getY());
                                             28
                                                                                        Figure 5
```

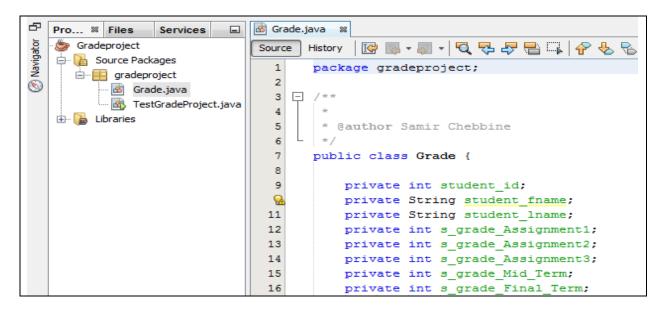
3. Using Array of Objects in GradeProject

- a) Create a Java Project to be named *GradeProject* using Eclipse IDE to define a new data structure type, called *Grade*.
- b) Create a class to define data structure type, called *Grade*, which is designed to group data and functions into a single unit that *represents* a template of the fields used in *Grade.in* as shown in the following Figure.
- c) Each line within *Grade.in* represents a Grade's record with the following fields: student_id (int), student_lname (string), student_fname (string), s_grade_Assignment1 (int), s_grade_Assignment2(int), s_grade_Assignment3(int), s_grade_Mid_Term (int), s_grade_Final_Term (int).
- d) Add **default constructor** (student_id=0, student_fname="", student_lname="", s_grade_Assignment1=0,s_grade_Assignment2=0, s_grade_Assignment3=0,

s_grade_Mid_Term=0, s_grade_Final_Term=0) and **constructor with parameters** (get the same name as the name of the class) within the *Grade* class in order to initialize the data members

(student_id,student_fname,student_lname,s_grade_Assignment1,s_grade_Assignment2, s_grade_Assignment3, s_grade_Mid_Term, s_grade_Final_Term) of every object.

- e) Add public **Mutator** (**setter**) methods in *Grade* class to modify the values of private members.
- f) Add public **Accessor** (**getter**) methods in *Grade* class to read the values of private members.
- g) Add a method called public String toString() in *Grade* class to print the *Publisher* information in the form of "S Id :"+ student_id +"S LName :"+ student_lname + "S FName :" + student_fname + "S Ass1:" + s_grade_Assignment1 + "S Ass2:" + s_grade_Assignment2 + "S Ass3:" + s_grade_Assignment3 + "S MT:" + s_grade_Mid_Term + "S FT:" + s grade Final Term.
- h) Add a method within *Grade* class called (Calculate_GradeAverage ()) that calculates and returns the average score for each student according to the following mark distribution:
 - 1) 40 % for all assignments (s_grade_Ass1 for Assignment 1), (s_grade_Ass2 for Assignment 2), (s_grade_Ass3 for Assignment 3)
 - 2) 30 % for Mid Term Exam (*s_grade_Mt*)
 - 3) 30 % for Final Exam (s grade Ft)



- i) You need to test the implemented class Grade in the main program TestGradeProject.
- j) Instantiate an array object of <code>Grade</code> class type to be referenced by (<code>all_sgrades</code>) of <code>Grade</code> class type) using the **default constructor**. Then read records from the input file <code>Grade.in</code> as shown in Figure 4 and assign the instance data of every record of the input File into its corresponding array object component using the implemented **setter** methods.
- k) Invoke the method Calculate_GradeAverage () related to every array object component, and display the student information using the **getter** methods, the grade average of every student, and the total grade average of all students as shown in the following Figure.

	□ Grade.in - Notepad							
<pre>Grade[] all sgrades = new Grade[6];</pre>	File Edit Format View Help							
	12 2 3 4 5 6	Graham Sanchez White Phelp Lewis James	Bill Jim Peter David Sheila Thomas	69 88 85 70 50 89	70 72 80 60 76 99	98 90 45 60 87 97	80 83 93 90 59 98	90 93 70 70 72 99
		Figure 4	4					

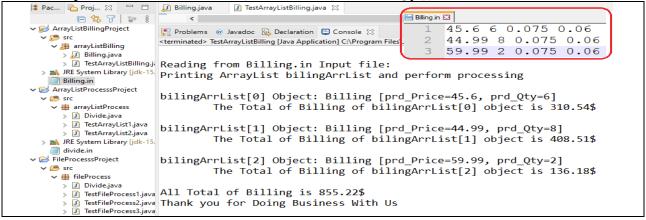
```
4) s_id: 4, Stdent Last Name: Phelp, Student First Name: David
Output - Gradeproject (run) 88
     1) s id: 0, Stdent Last Name: Graham, Student First Name: Bill
                                                                                       Mid Term Exam : 70
.
200
                                                                                      The Average score for this Student is 73.33
      Mid Term Exam : 80
Final Exam : 90
                                                                                      5) s_id: 5, Stdent Last Name: Lewis, Student First Name: Sheila
      The Average score for this Student is 82.60
      2) s_id: 2, Stdent Last Name: Sanchez, Student First Name: Jim
                                                                                      The Average score for this Student is 67.70
       Mid Term Exam
      The Average score for this Student is 86.13
                                                                                       Final Exam : 99
      3) s_id: 3, Stdent Last Name: White, Student First Name: Peter
                                                                                      The Average score for this Student is 97.10
       Assignment 2 : 80
Assignment 3 : 45
                                                                                      Number of Students in the file is 6
       Mid Term Exam : 93
       The Average score for this Student is 76.90
```

- 1) (Parallel Arrays) Add more Java statements in the main program to do the following:
 - Store the first column of the file *Grade.in* which represents the field student_id into *one-dimensional array* of *int* type, and print its components.
 - Store the second and third columns of the file *Grade.in* which represents the fields student_lname (string), student_fname (string) into *parallel one-dimensional array* of *String* type each, and print its components.
 - Store all column grades s_grade_Assignment1, s_grade_Assignment2, s_grade_Assignment3, s_grade_Mid_Term, s_grade_Final_Term of the file *Grade.in* into two dimensional array of double type each, and print its components.

4. Using ArrayList Data Structure in ArrayListBillingProject

Create a Java Project **ArrayListBillingProject** using NetBeans IDE that allows end user to issue a billing by populating data file Billing.in into an **ArrayList** data structure of type Billing class type.

- a) You need to design a **Java class** called **Billing**, which takes the price and quantity as two **private** non static members called respectively (prd_Price) and (prd_Qty). The variables called Fed_Tax, Prv_Tax as **public** and static data members.
 - 1) Add **default constructor**, setters, getters, and toString()
 - 2) Add a method called CalculateBilling() in Billing class to calculate the total of billing T_Billing = (prd_Price* prd_Qty) + (prd_Price*prd_Qty)* Fed_Tax + (prd_Price*prd_Qty)* Prv_Tax
- b) Create *TestArrayListBilling.java* where you populate an ArrayList data structure of Billing class type to be referenced by (billingArrList) from input file Billing.in. Set every component using the implemented setter methods (setPrd_Price(), setPrd_Qty()).



Calculate the total of billing of all ArrayList components. Display the total as shown above.