CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name **Student ID** **Point Total**

Section 1: Define / Answer

Describe what computer events that can be handled in java programming-

EVENT CLASS(Handlers)	LISTENER INTERFACE(Complete)
ActionEvent	ActionListener
ItemEvent	ItemListener
MouseEvent	MouseListener
KeyEvent	KeyListener
WindowEvent	WindowListener
ContainerEvent	ContainerListener
ComponentEvent	ComponentListener
FocusEvent	FocusListener
AdjustmentEvent	AdjustmentListener

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

Task 1:

USE OBJECT ORIENTATED PROGRAM DESIGN TO SOLVE PROBLEM

Update Assignment #14.

Create 1 "Programmer Created" Class to read student data from the text file, and create student objects. (Objects will need to be stored in a data structure)

Create a class Student. Containing First Name, Last Name, private DOB, Social Security Number.

Containing protected class variables Street Address, and Zip Code

Containing public Student ID number, Major.

The program should execute in way that student objects are created.

Create a graphical user interface with buttons to run the program.

Use Comparator Interface to sort over different data fields.

Create private modifiers for sensitive materials.

Return redacted versions of social security, DOB.

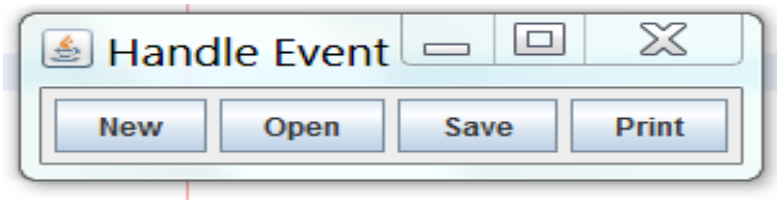
For example – Social security = XXX-XX-8010

DOB – XX/XX/1980

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

Attach Snipping Photos Below



If a button is pressed a sorted list of students by last name.

If a different button is pressed a sorted list of students by student ID.

etc.

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```
1
2   package interface3;
3
4   public class Address extends Student implements foundationforAddress {
5       protected String streetAddress, zipCode;
6
7       public Address(String firstName, String lastName,
8               String streetAddress, String zipCode){
9           super(firstName, lastName);
10          this.streetAddress = streetAddress;
11          this.zipCode = zipCode;
12      }
13
14      public void setstreetAddress(String streetAddress){
15          this.streetAddress = streetAddress;
16      }
17
18      @Override
19      public String getstreetAddress(){
20          return streetAddress;
21      }
22
23      public void setzipCode(String zipCode){
24          this.zipCode = zipCode;
25      }
26
27      @Override
28      public String getzipCode(){
29          return zipCode;
30      }
31
32      @Override
33      public void print(){
34          System.out.println("StreetAddress: " + getstreetAddress() +
35                  "\nZipCode: " + getzipCode());
36      }
37  }
38
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```
1
2   package interface3;
3
4   public class Info extends Address implements foundationforInfo{
5       String studentID, major;
6
7       public Info(String firstName, String lastName,
8               String streetAddress, String zipCode,
9               String studentID, String major){
10          super(firstName, lastName, streetAddress, zipCode);
11          this.studentID = studentID;
12          this.major = major;
13      }
14
15      @Override
16      public void print(){
17          System.out.println("FirstName: " + firstName +
18                  "\nLastName: " + lastName +
19                  "\nDate of Birth: XX-XX-" + getdob().substring(4) +
20                  "\nSSID: XXX-XX-" + getssid().substring(5) +
21                  "\nStreetAddress: " + getstreetAddress() +
22                  "\nZipCode: " + getzipCode() +
23                  "\nStudentID: " + studentID +
24                  "\nMajor: " + major);
25      }
26  }
27
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```
1
2   package interface3;
3
4   public class Student implements foundationforStudent {
5       String firstName, lastName;
6       private String dob, ssid;
7       public Student(String firstName, String lastName){
8           this.firstName = firstName;
9           this.lastName = lastName;
10      }
11
12      public void setdob(String dob){
13          this.dob = dob;
14      }
15
16      @Override
17      public String getdob(){
18          return dob;
19      }
20
21      public void setssid(String ssid){
22          this.ssid = ssid;
23      }
24
25      @Override
26      public String getssid(){
27          return ssid;
28      }
29
30      @Override
31      public void print(){
32          System.out.println("FirstName: " + firstName +
33                  "\nLastName: " + lastName +
34                  "\nDate of Birth: XX-XX-" + getdob().substring(4) +
35                  "\nSSID: XXX-XX-" + getssid().substring(5));
36      }
37  }
38
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package interface3;

interface foundationforStudent{
    public String getdob();
    public String getssid();
    public void print();
}

interface foundationforAddress{
    public String getstreetAddress();
    public String getzipCode();
    public void print();
}

interface foundationforInfo{
    public void print();
}
```

```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package interface3;

import java.util.Comparator;

public class LastNameComparator implements Comparator<Info> {

    @Override
    public int compare(Info o1, Info o2) {
        String name1 = o1.lastName;

        String name2 = o2.lastName;

        //Ascending name1.compareTo(name2)
        //Decending name2.compareTo(name1)
        return name1.toLowerCase().compareTo(name2.toLowerCase());
    }
}
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package interface3;

import java.util.Comparator;

public class StreetAddressComparator implements Comparator<Info> {
    @Override
    public int compare(Info o1, Info o2) {
        String name1 = o1.streetAddress;
        String name2 = o2.streetAddress;

        //Ascending name1.compareTo(name2)
        //Decending name2.compareTo(name1)

        return name1.toLowerCase().compareTo(name2.toLowerCase());
    }
}
```

```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package interface3;

import java.util.Comparator;

public class StudentIdComparator implements Comparator<Info>{

    @Override
    public int compare(Info o1, Info o2){

        //Ascending Integer.parseInt(o1.studentID) - Integer.parseInt(o2.studentID)
        //Descending Integer.parseInt(o2.studentID) - Integer.parseInt(o1.studentID)
        return Integer.parseInt(o1.studentID) - Integer.parseInt(o2.studentID);
    }
}
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package interface3;

import java.util.Scanner;
import java.util.Comparator;
import java.util.Arrays;
import java.util.Collections;


class Menu{

    Info[] arylist = new Info[10];
    java.io.File file = new java.io.File("StudentInfo.txt");

    public Menu(Info[] ary){
        this.arylist = ary;
    }

    public void menuDisplay(){
        System.out.println(
                        "\n*******************************************"+
                        "\n*          Main Menus                     *" +
                        "\n*Enter# to run program or Quit            *" +
                        "\n*1)Print Sorted List of Student Names     *" +
                        "\n*2)Print Sorted List of Student Address   *" +
                        "\n*3)Print Sorted List of Student ID number *" +
                        "\n*4)Quit                                  *" +
                        "\n*******************************************");
    }

    public void menu(){
        try{
            arylist = findList();
            Button first = new Button(arylist);
            menuDisplay();
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name **Student ID** **Point Total**

```
            first.gui(arylist);
        } catch (Exception e) {
            System.out.println("Invalid Input");
        }
    }


    public void write() throws Exception{
        file = new java.io.File("C:\\Users\\student\\Documents\\NetBeansProjects\\Interface3\\StudentInfo.txt");
        java.io.PrintWriter output = new java.io.PrintWriter(file);
        for(int i = 0; i < arylist.length; i++){
            output.println(arylist[i].firstName + " " + arylist[i].lastName + " " +
                    arylist[i].getdob() + " " + arylist[i].getssid() + " " +
                    arylist[i].getstreetAddress() + " " + arylist[i].getzipCode() + " " +
                    arylist[i].studentID + " " + arylist[i].major);
        }
        //assignemnt 14: when write, put the objective to the first
        // or limit to until next space

        output.close();

    }

    public void read() throws Exception{ //option for case 1: if lastname_1, if ID_2, ...
        java.io.File file = file = new java.io.File("C:\\Users\\student\\Documents\\NetBeansProjects\\Interface3\\StudentInfo.txt");
        String str;
        Scanner input = new Scanner(file);

        while(input.hasNext()){
            str = input.nextLine();
            System.out.println(str);
        }
    }


    public Info[] findList() throws Exception{
        java.io.File file = file = new java.io.File("C:\\Users\\student\\Documents\\NetBeansProjects\\Interface3\\StudentInfo.txt");
        String str;
        Scanner input = new Scanner(file);
        int i = 0;
           while(input.hasNext()){
               str = input.nextLine();
               String[] tokens = str.split(" ");

               for(int j = 0; j < 8; j++) {
                   if(j == 0){
                       arylist[i].firstName = tokens[j];
                   } else if(j == 1){
                       arylist[i].lastName = tokens[j];
                   } else if(j == 2){
                       arylist[i].setdob(tokens[j]);
                   } else if(j == 3){
                       arylist[i].setssid(tokens[j]);
                   } else if(j == 4){
                       arylist[i].streetAddress = tokens[j];
                   } else if(j == 5){
                       arylist[i].zipCode = tokens[j];
                   } else if(j == 6){
                       arylist[i].studentID = tokens[j];
                   } else if(j == 7){
                       arylist[i].major = tokens[j];
                   }
               }
               i++;
           }
           return arylist;
       }
   }
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```
1  /*
2   * To change this license header, choose License Headers in Project Properties.
3   * To change this template file, choose Tools | Templates
4   * and open the template in the editor.
5   */
6  package interface3;
7
8  import java.util.Arrays;
9  import javax.swing.*;
10 import java.awt.event.*;
11
12 public class Button extends JFrame {
13
14     public Button(Info[] arylist){
15         JButton sortName = new JButton("sortName");
16         JButton sortAddress = new JButton("sortAddress");
17         JButton sortId = new JButton("sortId");
18         JButton quit = new JButton("Quit");
19         JButton menu = new JButton("Menu");
20
21         JPanel panel = new JPanel();
22         panel.add(menu);
23         panel.add(sortName);
24         panel.add(sortAddress);
25         panel.add(sortId);
26         panel.add(quit);
27
28         add(panel);
29
31         sortName.addActionListener(new ActionListener() {
             @Override
             public void actionPerformed(ActionEvent e) {
33               System.out.println("---Before sorting---");
34               for(int i = 0; i < 10; i++){
35                   //ary[i].print();
36                   System.out.println(arylist[i].lastName);
37
38               }
39
40               Arrays.sort(arylist, new LastNameComparator());
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```
41                  System.out.println("---After sorting---");
42                  for(int i = 0; i < 10; i++){
43                      //ary[i].print();
44                      System.out.println(arylist[i].lastName);
45                  }
46              }
47          });
48
            sortAddress.addActionListener(new ActionListener() {
50              @Override
                public void actionPerformed(ActionEvent e) {
52                  System.out.println("---Before sorting---");
53                  for(int i = 0; i < 10; i++){
54                      //ary[i].print();
55                      System.out.println(arylist[i].streetAddress);
56
57                  }
58
59                  Arrays.sort(arylist, new StreetAddressComparator());
60                  System.out.println("---After sorting---");
61                  for(int i = 0; i < 10; i++){
62                      //ary[i].print();
63                      System.out.println(arylist[i].streetAddress);
64                  }
65              }
66          });
67
            sortId.addActionListener(new ActionListener() {
69              @Override
                public void actionPerformed(ActionEvent e) {
71                  System.out.println("---Before sorting---");
72                      for(int i = 0; i < 10; i++){
73                          //ary[i].print();
74                          System.out.println(arylist[i].studentID);
75
76                      }
77
78                      Arrays.sort(arylist, new StudentIdComparator());
79                      System.out.println("---After sorting---");
80                      for(int i = 0; i < 10; i++){
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**

```
81                          //ary[i].print();
82                          System.out.println(arylist[i].studentID);
83                      }
84                  }
85              });
86
87              quit.addActionListener(new ActionListener() {
88                  @Override
89                  public void actionPerformed(ActionEvent e) {
90                      System.out.println("You quited the program.");
91                      System.exit(0);
92                  }
93              });
94
95
96              menu.addActionListener(new ActionListener() {
97                  @Override
98                  public void actionPerformed(ActionEvent e) {
99                      Menu first = new Menu(arylist);
100                     first.menuDisplay();
101                 }
102             });
103
104
105         }
106
107         public void gui(Info[] arylist){
108             JFrame sample = new Button(arylist);
109             sample.setTitle("Main Menu");
110             sample.pack();
111             sample.setLocationRelativeTo(null);
112             sample.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
113             sample.setVisible(true);
114         }
115     }
116
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

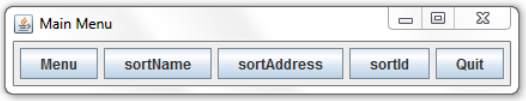
Student Name _____ **Student ID** _____ **Point Total** _____

```
1
2   package interface3;
3
4   import java.util.Scanner;
5   import java.util.Comparator;
6   import java.util.Arrays;
7   import java.util.Collections;
8
9   public class Operator {
10
11      public static void main(String[] args)throws Exception{
12          Info[] ary = new Info[10];
13          ary[0] = new Info("Kachi", "Lau", "Oakland", "94612", "10819338", "CS");
14          ary[0].setdob("01081993");
15          ary[0].setssid("123456789");
16          ary[1] = new Info("Jacky", "Chan", "SanDeigo", "94111", "10719922", "Math");
17          ary[1].setdob("07021992");
18          ary[1].setssid("888888888");
19          ary[2] = new Info("Tank", "Lam", "SanFranscio", "94512", "10325361", "CS");
20          ary[2].setdob("02021997");
21          ary[2].setssid("111111111");
22          ary[3] = new Info("Kitty", "Lu", "Oakland", "12354", "12345678", "Physic");
23          ary[3].setdob("03031988");
24          ary[3].setssid("777777777");
25          ary[4] = new Info("Ken", "chang", "SanFrancisco", "94512", "10232153", "CS");
26          ary[4].setdob("04041987");
27          ary[4].setssid("222222222");
28          ary[5] = new Info("Ryu", "Kawasaki", "Oakland", "94612", "15123524", "CS");
29          ary[5].setdob("12311993");
30          ary[5].setssid("234567890");
31          ary[6] = new Info("Alex", "Taco", "Oakland", "94612", "21231523", "Math");
32          ary[6].setdob("07071996");
33          ary[6].setssid("579134628");
34          ary[7] = new Info("Chicken", "Chicken", "USA", "12325", "12314823", "CS");
35          ary[7].setdob("01011991");
36          ary[7].setssid("264831597");
37          ary[8] = new Info("Mc", "donald", "Oakland", "94612", "21353262", "CS");
38          ary[8].setdob("02031995");
39          ary[8].setssid("791346528");
40          ary[9] = new Info("Coffe", "Cup", "Oakland", "94612", "12381234", "CS");
41          ary[9].setdob("08081998");
42          ary[9].setssid("231535648");
43
44          Menu first = new Menu(ary);
45          first.write();
46          first.menu();
47      }
48  }
49
```

CIS 36B – 16th Class / Lab Assignment – **10 Points-**

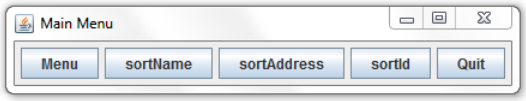
Student Name _____ **Student ID** _____ **Point Total**

```
run:

*****************************************
*          Main Menus                   *
*Enter# to run program or Quit          *
*1)Print Sorted List of Student Names    *
*2)Print Sorted List of Student Address  *
*3)Print Sorted List of Student ID number *
*4)Quit                                 *
*****************************************
---Before sorting---
Lau
Chan
Lam
Lu
chang
Kawasaki
Taco
Chicken
donald
Cup
---After sorting---
Chan
chang
Chicken
Cup
donald
Kawasaki
Lam
Lau
Lu
Taco

*****************************************
*          Main Menus                   *
*Enter# to run program or Quit          *
*1)Print Sorted List of Student Names    *
*2)Print Sorted List of Student Address  *
*3)Print Sorted List of Student ID number *
*4)Quit                                 *
*****************************************
---Before sorting---
SanDeigo
SanFrancisco
USA
Oakland
Oakland
Oakland
SanFranscio
Oakland
```



CIS 36B – 16th Class / Lab Assignment – **10 Points-**

Student Name _____ **Student ID** _____ **Point Total**




```
Oakland
Oakland
---After sorting---
Oakland
Oakland
Oakland
Oakland
Oakland
Oakland
SanDeigo
SanFrancisco
SanFranscio
USA

*****************************************
*          Main Menus                   *
*Enter# to run program or Quit          *
*1)Print Sorted List of Student Names    *
*2)Print Sorted List of Student Address  *
*3)Print Sorted List of Student ID number *
*4)Quit                                 *
*****************************************
---Before sorting---
12381234
21353262
15123524
10819338
12345678
21231523
10719922
10232153
10325361
12314823
---After sorting---
10232153
10325361
10719922
10819338
12314823
12345678
12381234
15123524
21231523
21353262

*****************************************
*          Main Menus                   *
*Enter# to run program or Quit          *
*1)Print Sorted List of Student Names    *
*2)Print Sorted List of Student Address  *
*3)Print Sorted List of Student ID number *
*4)Quit                                 *
*****************************************
You quited the program.
BUILD SUCCESSFUL (total time: 1 minute 9 seconds)
```