

Ahmadu Bello University  
 Department of Computer Science  
 Second Semester Examinations – August 2017  
 COSC212: Object Oriented Programming II – Marking Scheme

Attempt **FOUR** questions

**Marking Scheme**

Time: **120** mins

1. (a). **[12 Marks]**. Suppose a call to the `sortList(int array[])` method from the code below is made with values {5, 2, 9, 4, 1, 3, 6, 0} write in the table below each pass of the value *i*.

```
public static void sortList(int array[], Comparator c){
    int temp;
    for(int i = 0; i<array.length;i++){
        int min_index = i;
        for(int j = i+1; j< array.length; j++){
            if(c.compare(array[j], array[min_index])<0)
                min_index = j;
        }
        temp = array[i];
        array[i] = array[min_index];
        array[min_index] = temp;
        System.out.println(i+" "+Arrays.toString(array));
    }
}
```

**Solution**

Pass i = 0	0	2	9	4	1	3	6	5
Pass i = 1	0	1	9	4	2	3	6	5
Pass i = 2	0	1	2	4	9	3	6	5
Pass i = 3	0	1	2	3	9	4	6	5
Pass i = 4	0	1	2	3	4	9	6	5
Pass i = 5	0	1	2	3	4	5	6	9
Pass I = 6	0	1	2	3	4	5	6	9
Pass I = 7	0	1	2	3	4	5	6	9

(b). [8 Marks]. Examine the program source code below and write the outputs:

```
import java.util.Arrays;
public class SearchClass {
    public static void main(String args[]) throws Exception {
        int array[] = { 2, 5, -2, 6, -3, 8, 0, -7, -9, 4 };
        Arrays.sort(array);
        printArray("Sorted array", array);
        int index = Arrays.binarySearch(array, -3);
        System.out.println("Found -7 @ " + index);
    }
    private static void printArray(String message, int array[]){
        System.out.println(message + ": [length: " + array.length +
        "]" );

        for (int i = 0; i < array.length; i++) {
            if(i != 0) {
                System.out.print(", ");
            }
            System.out.print(array[i]);
        }
        System.out.println();
    }
}
```

Solution:

Sorted array : [length: 10]

-9, -7, -3, -2, 0, 2, 4, 5, 6, 8

Found -7 @ 2

2. [20 Marks]. Study the following program source code and answer questions i-vi below.

```
import java.util.*;
public class Book {
    private String author;
    private String title;
    private int bookNo;
    ArrayList<Book> books;
    static ListIterator it;
    public Book(String author, String title, int bookNo){
        this.author = author;
        this.title = title;
        this.bookNo = bookNo;
    }
    public Book(){
        books = new ArrayList<Book>();
        it = books.listIterator();
    }
    public String getAuthor(){
        return author;
    }
    public String getTitle(){
        return title;
    }
    public int getBookNo(){
        return bookNo;
    }
    public void setAuthor(String newAuthor){
        if(!newAuthor.equals(""))
            author = newAuthor;
    }
    public void setTitle(String newTitle){
        if(!newTitle.equals(""))
            title = newTitle;
    }
    public void setBookNo(int newBookNo){
        if(newBookNo > 0)
            bookNo = newBookNo;
    }
    public String toString(){
        return String.format("Author: %s\nTitle: %s\nBook
No.: %05d",author,title,bookNo);
    }
    public static void main(String[] args){
        Book bb = new Book();
        // methods calls
    }
} // end of class Book
```

i. [4 Marks]. Write a static method createBook( ) that will create two books and add them to ArrayList as declared in the above program.

```
public static void createBook(){
    Book book = new Book();
    book.books.add(new Book("Othman Aliyu","Java How to
Program",00001));
    book.books.add(new Book("Aisha Aliyu","Object Oriented
Programming in C",00002));
}
```

- ii. [3 Marks] Write a static method `showAllBooks()` that will display all the books added to the `ArrayList`.

```
public static void ShowAllBooks(){
    while(it.hasNext()){
        System.out.println(it.next());
    }
}
```

- iii. [3 Marks] Write a static method `nextBook()` that will display the next book from the `ArrayList`.

```
public static void nextBook(){
    if(it.hasNext()){
        System.out.println(it.next());
    }
}
```

- iv. [3 Marks] Write a static method `previousBook()` that will display the previous book from the `ArrayList`.

```
public static void previousBook(){
    if(it.hasPrevious()){
        System.out.println(it.previous());
    }
}
```

- v. [4 Marks] Write a static method `updateBook()` that will change the authors of the books to "Ahmadu Bello University, Zaria." for any bookNo greater than 10

```
public static void updateBook(){
    while(it.hasNext()){
        Book book = (Book)it.next();
        if(book.getBookNo()>10)
            book.setAuthor("Ahmadu Bello University");
    }
}
```

- vi. [4 Marks] Write a static method `removeBook()` that will remove all the books with `bookNo` less than 0 from the `ArrayList`.

```
public static void removeBook(){
    while(it.hasNext()){
        Book book = (Book)it.next();
        if(book.getBookNo() < 0)
            books.remove();
    }
}
```

3. [20 Marks] Write a GUI class (`Account.java`) that will display the window shown in Figure 1.1. The window (`JFrame`) has a fixed size of 500 by 200 pixels. Create a menu called **File** with two items namely **Open** and **Save**.

The first row has a label (`JLabel`) and a text box (`JTextField`) for which it should be wide enough for at least ten characters. The second row has a label (`JLabel`) and two radio buttons (`JRadioButton`) marked **Male** and **Female**, the third row has a label (`JLabel`) and a text area (`JTextArea`) with 5 rows and 10 columns, and the last row has two buttons (`JButton`), one marked **Clear** which will clear the contents of the text field and text area. The other button marked **Create Account** which will display the text from the text field in the text area.

**Note:** the two radio buttons for gender should be mutually exclusive

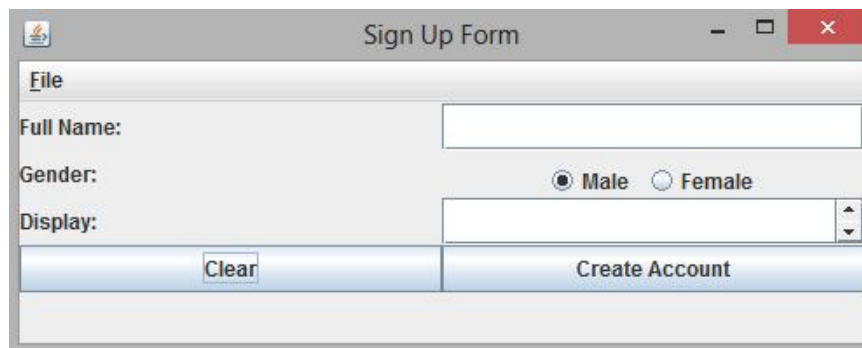


Figure 1.1

```

import java.awt.event.*;
import java.awt.*;
import javax.swing.*;
public class Account extends JFrame{
    private JButton btnCreate;
    private JButton btnClear;
    private JTextField txtName;
    private JTextArea txtDisplay;
    public Account(){
        super("Sign Up Form");
        JMenuBar menuBar = new JMenuBar();
        JMenu filemenu = new JMenu("File");
        filemenu.setMnemonic('F');
        JMenuItem open = new JMenuItem("Open");
        JMenuItem save = new JMenuItem("Save");
        filemenu.add(open);
        filemenu.add(save);
        menuBar.add(filemenu);
        setJMenuBar(menuBar);
        add(new JLabel("Full Name: "));
        add(txtName = new JTextField(10));
        add(new JLabel("Gender:"));
        JPanel panel = new JPanel();
        JRadioButton rbMale;
        JRadioButton rbFemale;
        panel.add(rbMale = new JRadioButton("Male",true));
        panel.add(rbFemale = new JRadioButton("Female"));
        ButtonGroup grp = new ButtonGroup();
        grp.add(rbFemale);
        grp.add(rbMale);
        add(panel);
        add(new JLabel("Display: "));
        add(txtDisplay= new JTextArea(5,10));
        JScrollPane scroll = new JScrollPane(txtDisplay);
        add(scroll);
        btnClear = new JButton("Clear");
        add(btnClear);
        btnCreate = new JButton("Create Account");
        add(btnCreate);
        setLayout(new GridLayout(5,2));
        setSize(500,200);
        setResizable(true);
        setVisible(true);
        btnCreate.addActionListener(new Handler());
        btnClear.addActionListener(new Handler());
    } // end constructor
    public class Handler implements ActionListener{
        public void actionPerformed(ActionEvent ae){
            if(ae.getSource() == btnClear){
                txtName.setText("");
                txtDisplay.setText("");
            }
            if(ae.getSource() == btnCreate){
                txtDisplay.setText(txtName.getText());
            }
        }
    }
    public static void main(String[] args){
        Form Account = new Account();
        form.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }
}

```