PECANWOOD

COLLEGE

Prepared for Life

INFORMATION TECHNOLOGY PRACTICAL JULY 2019
GRADE 10

NAME: With Memo _____ **GRADE:** _____

DATE: 12 JULY 2019
MODERATOR: MR N NAINAR
EXAMINER: MR SC EILERTSEN

MARKS: 85
TIME: 120 MINUTES

INSTRUCTIONS:

1. This exam is made up of 7 pages.
2. There is an addendum of 2 pages – "Addendum One"
3. You may use a non-programmable calculator where relevant.
4. Login immediately and load the IDE you intend to use.
5. Print your Java code for section two and three and hand it in with this question paper. You are responsible to ensure that your work is labelled correctly and handed in.
6. You must print both your UI class and template class – **both** must have your name and class featured in the comments section.
7. Note that there are no trick questions in this exam e.g. there are no deliberately missing semi colons, full stops or irrelevant spelling errors in the provided code examples – simple answer the question as stated.

Section One **Java – short questions.**

1.1) Answer the questions that followed after **carefully** studying the code snippet below.

Consider this code from a single class program

```
 5 public class RandomFlipper
 6 {
 7
 8    public static void main (String[]args)
 9    {
10       double myRandomDouble = Math.random();
11       System.out.println(myRandomDouble);
12
13       int myRandomInt = (int)(myRandomDouble);
14       System.out.println(myRandomInt);
15
16    }
17 }
```

1.1.1) Explain in detail what line 10 does. Generates a random number between zero and one (not including one) This double value is stored in the variable called myRandomDouble ✓ (3)

1.1.2) If the output for line 11 was the value 0.3886471590730, what value would line 14 output to the console?

0 zero ✓ (1)

1.2) Answer the question that follows after **carefully** studying the code snippet below.

```
 5 public class RandomFlipper
 6 {
 7
 8    public static void main (String[]args)
 9    {
10       double myRandomDouble = Math.random();
11       System.out.println(myRandomDouble);
12
13       int myRandomInt = (int)(myRandomDouble);
14       System.out.println(myRandomInt * 100);
15
16    }
17 }
```

1.2.1) If the output for line 11 was the value 0.3886471590730, what value would line 14 output to the console?

0 zero ✓ (0 * 100 = 0) (1)

1.3) Answer the question below after **carefully** studying the code snippet.

```
 5 public class RandomFlipper
 6 {
 7
 8    public static void main (String[]args)
 9    {
10       double myRandomDouble = Math.random();
11       System.out.println(myRandomDouble);
12
13       int myRandomInt = (int)(myRandomDouble * 100);
14       System.out.println(myRandomInt);
15
16    }
17 }
```

1.3.1) If the output for line 11 was the value 0.3886471590730, what value would line 14 output to the console?

38 ✓ (does not round) ✓ (1)

1.4) Answer the question below after **carefully** studying the code snippet.

```
 8    public static void main (String[]args)
 9    {
10       double myRandomDouble = Math.random();
11       System.out.println(myRandomDouble);
12
13       int myRandomInt = (int)(myRandomDouble * 100);
14       System.out.println(myRandomInt);
15
16       for (int i = 0; i < myRandomInt; i++)
17       {
18          System.out.print(i + " ");
19
20       }
21
22    }
23 }
```

1.4.1) If the output for line 11 was the value 0.1486471590730, what would be the output of the loop in line 16 to line 20 be?

0 1 2 3 4 5 6 7 8 9 10 11 12 13 ✓ (3)

numbers ✓
spaces ✓
on one line ✓

1.5) Answer the questions that followed after studying the code below.

Consider the code below from two different classes that are part of the same package (not all the code is shown to save space) (the line numbers are for reference only as the two classes are each contained in two different files)

```
 1 public static void main(String[]args)
 2 {
 3
 4    int firstInt = 5, secondInt = 8, answer;
 5
 6    AcceptIntegers myIntegers = new AcceptIntegers();
 7    .
 8    .
 9    .                                          5       8
10    . answer = myIntegers.addIntegers(firstInt, secondInt);
11    .
12
13 public class AcceptIntegers
14 {
15    int first, second, total;
16                                  5    8
17    public int addIntegers (int f, int s)
18    {
19    5 first = f;
20    8 second = s;
21       total = first + second;
22       return total;
23    }
```

1. 5.1) What is the value of "answer" in line 10?

13 ✓ ✓ (2)

1.5.2) Explain fully how lines 10, line 17 and line 22 all work together

Line 10 passes two integer arguments ie 5 and 8 ✓
Line 17 accepts the two integer parameters ie 5+8 ✓
Line 22 passes the calculated value ✓
back to the calling statement ✓
ie line 10. (4)

Total for this question: (15)

2.1) Code a two-class program that calculates the cost of tickets for a music concert. There are adult and children tickets. Tickets can be collected at no extra cost or they can be couriered for an additional cost.

The total cost of the tickets is . . .

- the number of adults multiplied by the cost of the adult tickets

added to . . .

- the number of children tickets multiplied by the cost of the children tickets.

If the person replies that they would like to use the courier service, then the cost of the delivery must be added to the cost of the tickets – this gives us the **final cost.**

- The cost of the adult tickets are R150-50.
- The cost of the children tickets are R100.50
- The courier fees are R125-00.

These prices must not accepted via the keyboard. They must be hard coded into the ConcertUI class.

You have been supplied with the ConcertUI class that has the main method. There are two copies i.e. a hard copy for you to study to be found in Addendum One of this exam paper. You have also been supplied with the java file that you must use to answer this question.

Take careful note of the **use of comments** throughout the supplied ConcertUI class – these comments are part of the exam question, and part of the exam paper, and must be read accordingly.

2.1.1) **Write the template class that accompanies the ConcertUI class**. Your class must do the processing. This class must have **two methods**. These two methods must calculate the **total ticket price** and the **final price** respectively. Study ConcertUI for more detail. (24)

You may alter the ConcertUI class to suit your needs although this is not recommended.

Input – ConcertUI - Supplied
Study Addendum One as supplied. Note of the use of comments throughout the ConcertUI class.

Processing – Name this to match the ConcertUI class.

- Two different methods. Name these methods to match the ConcertUI class.
- One method determines the total cost of the tickets.
- The second method adds the courier fees if relevant – the final price.

Output - ConcertUI
Study Addendum One as supplied. Your template class must return the values needed so that the output lines 80 to 97 work correctly. Note that sample output is supplied with Addendum One.

Printing
Print both the ConcertUI class and your own template class that you have written.

NOTE: Credit is given for good programming practice i.e.

1. Well-chosen variable names and correctly created package and class names.
2. Indentation and Comments.
3. Good use of whitespace to group code into useful sections. (4)

Total for this question: (28)

Section Three **Java Coding – Object Orientated Programming – Creating a two class program**

3.1) Code a two-class program that a teacher can use to calculate both the student's term mark as well as determine the student's symbol.

The **term marks** are made up as follows (together they total 100)

- The cycle test marked out of 50
- A project marked out of 25
- A class test marked out of 15
- Another class test marked out of 10

The **symbols** allocated are as follows . . .

- A+ - Above 90
- A – Between 80 and 89
- B – Between 70 and 79
- C – Between 60 and 69
- D – Between 50 and 59
- Below 50. A special message must appear telling the student that they must submit another project for evaluation.

Two Class Program
Your program must comprise of a UI class that handles input and output, as well as another class that does the processing.

Input – UI Class
Accepts the student's name as well as their marks for the cycle test, the project and two class tests as outlined above. (27)

Processing – A template class

- Two different methods.
    - The first method must add all the marks together.
    - The second method must determine the student's symbol. (14)

Output – UI Class
Your template class must return the values and information to the UI class which must output the information to screen.

(TURN OVER FOR MORE) . . .

Sample Output
Your output should look something like this.

Example One
Student: Luke Skywalker
==========
Cycle test: 45
Project: 19
Class Test 1: 10
Class Test 2: 5
==========
Total: 79
==========
Symbol: B

Example Two
Student: Ben Kenoby
==========
Cycle test: 23
Project: 8
Class Test 1: 9
Class Test 2: 8
==========
Total: 48
==========
Symbol: Ben Kenoby must submit an additional project for evaluation.

Printing
Print both the UI class and your own template class with your name and grade in the comments section.

NOTE: Credit is given for good programming practice i.e.
1. Well-chosen variable names and correctly created package and class names.
2. Indentation and Comments.
3. Good use of whitespace to group code into useful sections. (3)

Total for this question: (44)

TOTAL: 85

Addendum One. With Memo

```
1  /* A two class program that calculates the cost of tickets for a music concert.
2  * There are adult and children tickets. Tickets can be collected for no extra cost
3  * or they can be couriered for an additional cost.
4  */
5
6  package julypracticalexam;
7
8  import javax.swing.JOptionPane;
9
10 public class ConcertUI
11 {
12    public static void main (String[]args)
13    {
14       //DECLARE AND INITIALISE VARIABLES AND OBJECTS
15       TicketCost myTicketCost = new TicketCost();
16
17       int numberOfAdultTickets = 0, numberOfChildrenTickets = 0;
18
19       String numberOfAdultTicketsSt, numberOfChildrenTicketsSt;
20       String courierSt;
21
22       double costOfAdultTickets = 150.50; //R150-00
23       double costOfChildrenTickets = 100.50; // R100-50
24       double costOfCourierFees = 120.25; // R120-25
25       double totalCostOfTickets = 0.0; // Tickets only
26       double finalCost = 0.0; // Tickets plus courier fees if relevant
27       boolean courier = false; // courier is needed, true or false
28
29       // INPUT
30       numberOfAdultTicketsSt = JOptionPane.showInputDialog("Number of adult tickets?");
31       numberOfChildrenTicketsSt = JOptionPane.showInputDialog("Number of children tickets?");
32       courierSt = JOptionPane.showInputDialog("Courier tickets? Y/N");
33
34       // Format the answer to courier correctly ie Y or N only
35       // Ensure that courierSt is only one character long.
36       courierSt = courierSt.substring(0,1);
37       // Ensure that courierSt is a capital letter
38       courierSt = courierSt.toUpperCase();
39
40       numberOfAdultTickets = Integer.parseInt(numberOfAdultTicketsSt);
41       numberOfChildrenTickets = Integer.parseInt(numberOfChildrenTicketsSt);
42
43       if (courierSt.equals("Y"))
44       {
45          courier = true;
46          System.out.println("Add courier fees");
47          System.out.println("");
48       }
49       else if (courierSt.equals("N"))
50       {
51          courier = false;
52          System.out.println("Do not add courier fees");
53          System.out.println("");
54       }
55       else
56       {
57          System.out.println("Error with courier fee input");
58          System.exit(0); // Program stops running due to error.
59       }
60
61
62
```

Addendum One (cont)

```
63
64        //PROCESSING
65        // Total cost of the tickets is worked out
66        totalCostOfTickets = myTicketCost.ticketCost(numberOfAdultTickets, numberOfChildrenTick
ets, costOfAdultTickets, costOfChildrenTickets);
67
68        // The courier fees are added if relevant
69        if (courier == true)
70        {
71           finalCost = myTicketCost.ticketCourier(totalCostOfTickets, costOfCourierFees);
72        }
73        else
74        {
75           finalCost = totalCostOfTickets; // No courier fees
76        }
77
78        // OUTPUT
79        System.out.println("Number of adults tickets is " + numberOfAdultTickets);
80        System.out.println("Number of children tickets is " + numberOfChildrenTickets);
81
82        if (courier == true)
83        {
84           System.out.println("Courier fees apply");
85        }
86        else
87        {
88
89           System.out.println("Courier fees do not apply");
90        }
91
92        System.out.println("Total cost of tickets is R" + totalCostOfTickets);
93        System.out.println("The final cost is R" + finalCost); // With courier fees
94     }
95  }
```

SAMPLE OUTPUT

```
1
2     ----jGRASP exec: java ConcertUI
3     Add courier fees
4
5     Number of adults tickets is 10
6     Number of children tickets is 20
7     Courier fees apply
8     Total cost of tickets is R3515.0
9     The final cost is R3635.25
10
11    ----jGRASP: operation complete.
12
13    ----jGRASP exec: java ConcertUI
14    Do not add courier fees
15
16    Number of adults tickets is 20
17    Number of children tickets is 10
18    Courier fees do not apply
19    Total cost of tickets is R4015.0
20    The final cost is R4015.0
21
22    ----jGRASP: operation complete.
23 ▶▶
```

sample one

sample two

xxx Section two Grade 10 July practical exam

```
1  // comment
2
3  package julypracticalexam;
4
5  public class TicketCost
6  {
7      private int numberOfAdults, numberOfChildren;
8      private double costOfAdultTickets, costOfChildrenTickets;
9      private double costOfCourierFee = 0.0;
10     private double totalCostOfTickets = 0.0;
11     private double finalCost = 0.0;
12
13     public double ticketCost(int na, int nc, double ca, double cc)
14     {
15         numberOfAdults = na;
16         numberOfChildren = nc;
17         costOfAdultTickets = ca;
18         costOfChildrenTickets = cc;
19
20         totalCostOfTickets = (numberOfAdults * costOfAdultTickets) + (numberOfChildren
 * costOfChildrenTickets);
21
22         return totalCostOfTickets;
23
24     } // end ticketCost
25
26     public double ticketCourier(double tc, double cf)
27     {
28         totalCostOfTickets = tc;
29         costOfCourierFee = cf;
30
31         finalCost = totalCostOfTickets + costOfCourierFee;
32
33         return finalCost;
34
35     }
36
37 }
```

must match.

must match

must match

must match.

Calc = (4)

x/24

Variables + class names ✓

Indentation + comments ✓

White space ✓

x/27

```
1 // Section Three. Grade 10 July practical exam
2
3 package sectionthree;
4
5 import javax.swing.JOptionPane;
6
7 public class TeacherCalculatorUI
8 {
9
10    public static void main (String[]args)
11    {
12       // DECLARATIONS
13       String studentName;
14       int cycleTest, project, classTest1, classTest2, total;
15       String symbol;
16
17       TeacherCalculator myTeacherCalculator = new TeacherCalculator();
18
19       // INPUT
20       studentName = (JOptionPane.showInputDialog("Enter the student's name"));
21       cycleTest = Integer.parseInt(JOptionPane.showInputDialog("Enter the cycle test mark out
 of 50"));
22       project = Integer.parseInt(JOptionPane.showInputDialog("Enter the project mark out of 2
5"));
23       classTest1 = Integer.parseInt(JOptionPane.showInputDialog("Enter class test 1 mark out
of 15"));
24       classTest2 = Integer.parseInt(JOptionPane.showInputDialog("Enter class test 2 mark out
of 10"));
25
26       // PROCESSING
27       total = myTeacherCalculator.calculateMark(cycleTest, project, classTest1, classTest2);
28       symbol = myTeacherCalculator.determineSymbol(total);
29
30       // OUTPUT
31       System.out.println("Student:" + studentName);
32       System.out.println("====================");
33       System.out.println("Cycle Test:" + "\t" + cycleTest);
34       System.out.println("Project:" + "\t" + project);
35       System.out.println("Class Test1:" + "\t" + classTest1);
36       System.out.println("Class Test2:" + "\t" + classTest2);
37       System.out.println("====================");
38       System.out.println("Total:" + "\t" + total);
39       System.out.println("====================");
40
41       if (total >= 50)
42       {
43          System.out.println("Symbol: " + symbol);
44       }
45
46       else if (total <= 49)
47       {
48          System.out.println(studentName + " " + symbol);
49       }
50
51    }
52 }
53
```

Layout (4)

logic for alternative message (4)

```java
// Section Three. Grade 10 July practical exam

package sectionthree;

public class TeacherCalculator
{

    int cycleTest, project, classTest1, classTest2, total = 0;
    String symbol;

    public int calculateMark(int c, int p, int ct1, int ct2)
    {
      cycleTest = c;
      project = p;
      classTest1 = ct1;
      classTest2 = ct2;

      total = cycleTest + project + classTest1 + classTest2;

      return total;
   } // end calculateMark

   public String determineSymbol (int total)
   {
      if (total >=90)
         symbol = "A+";

      else if (total >= 80 && total <= 89)
         symbol = "A";

      else if (total >= 70 && total <= 79)
         symbol = "B";

      else if (total >= 60 && total <= 69)
         symbol = "C";

      else if (total >= 50 && total <= 59)
         symbol = "D";
      else if (total <=49)
         symbol = " must submit an additional project for marking.";

      return symbol;

   } // end determineSymbol

}
```

$\frac{14}{15}$

Variables + class names ✓

Indentation + comments ✓

Whitespace ✓

$\frac{3}{3}$