education

Department:
Education
PROVINCE OF KWAZULU-NATAL

INFORMATION TECHNOLOGY

PAPER 1

TRIAL EXAMINATION 2016

GRADE 12

DATE: 30 August 2016

MARKS: 150

TIME: 3 hours

EXAMINER : Ms. S BENIMADHO
MODERATOR : Mr. A J PILLAY

INSTRUCTIONS AND INFORMATION

1. Answer **ALL** the questions.
2. The duration of this examination is three hours. Because of the nature of this examination it is important to note that you will not be permitted to leave the examination room before the end of the examination session.
3. Make sure that you answer the questions according to the specifications that are given in each question. Marks will only be awarded based on the set requirements.
4. Only answer what is asked in each question. For example, if the question does not ask for data validation, then no marks will be awarded for data validation.
5. Your programs must be coded in such a way that it will work with any data and not just the sample data supplied or any data extracts that appear in the question paper.
6. **Make sure that you develop routines, such as search, sort and selection, from first principles and not use the built-in features of a programming language for any of these routines.**
7. An ASCII table is supplied on page 12, if required.
8. Save your work regularly on the disk (CD/flash disk/DVD, et cetera) that you have been given, or on the disk space allocated to you for this examination.
9. Make sure that your **name and surname** appears as a comment in the first line of code. Also include the question number as part of the comment.
10. At the end of this examination session, you must hand in the disk/CD with all your work saved on it OR you must make sure that all your work has been saved on the disk space allocated to you. Ensure that all files can be read.
11. You have been provided with the following Netbeans Projects, jFrame Forms and Text files:

Project	Question1	Question2	Question3
JFrame	question1	question2	question3
Java Class		question2Object	
Text File	GiftCodes	EmployeeDetails	

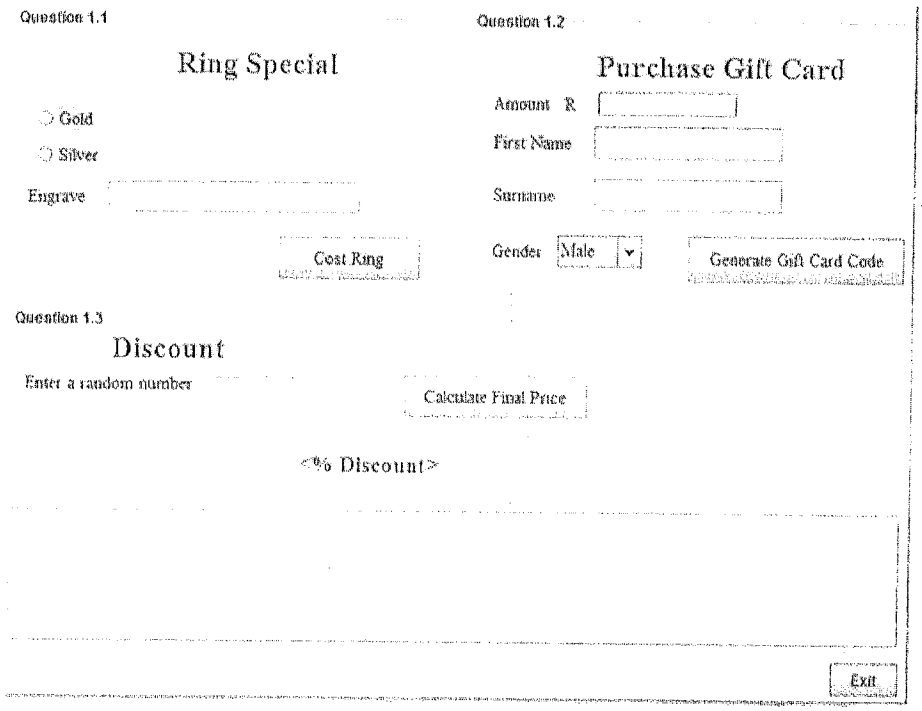
12. Print ALL coding done and complete the folder provided.
13. Password : **CiPu*$T16**

SCENARIO

CIPU Jewelry's has many stores around Kwa Zulu Natal. Each store has two managers and other staff of various different job titles such as; salesperson, administration staff, Jewelry designer, general cleaners etc. The following programs need to be created to assist the CEO of CIPU Jewelry's.

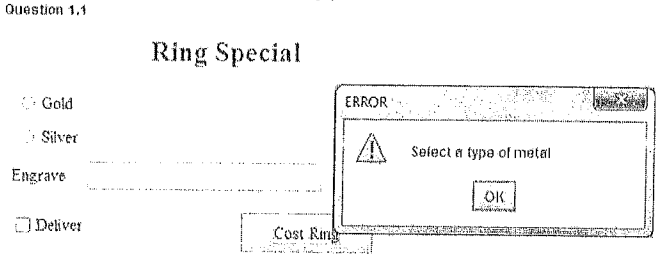
QUESTION 1

Basic Gold and Silver band rings are on sale. Open the jFrame question1, in the project Question1.



1.1 Code the button ***btnCostRing***.

1.1.1 Allow the user to select the metal for the ring, either Gold or Silver. If neither metal has been selected, a **Warning Message** dialog box must be displayed stating : "Please select a metal type.". (3)



1.1.2 The standard cost of one ring is as follows:

Type of Metal	Cost per Ring
Gold	R300.00
Silver	R200.00

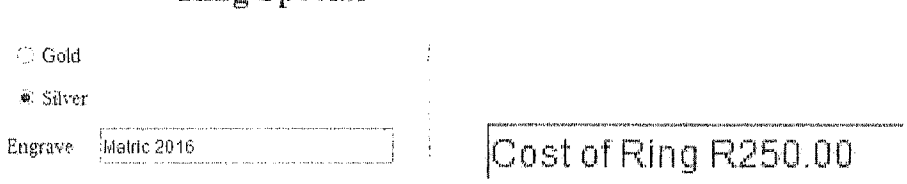
Calculate the cost of the selected ring. (4)

1.1.3 The ring purchased could be engraved, however NO more than fifteen characters are allowed. If more than fifteen characters have been entered in the text field, *txfEngrave*, the text field must be set back to empty. (4)

If the user has entered fifteen characters or less, a charge of R5.00 per character to be **engraved** is charged. (6)

1.1.4 Display in the text area, *txaDisplay*, the total cost of the ring, rounded off to two decimal places.

Question 1.1



(2)

1.2 Code the button ***btnGiftCardCode***.

1.2.1 Extract the gift card amount, first name, surname and gender, from the following components; *txfAmountGiftCard, txfFirstName, txfSurname* and *cmbxGender*. (3)

1.2.2 Generate a gift card code using the following criteria:

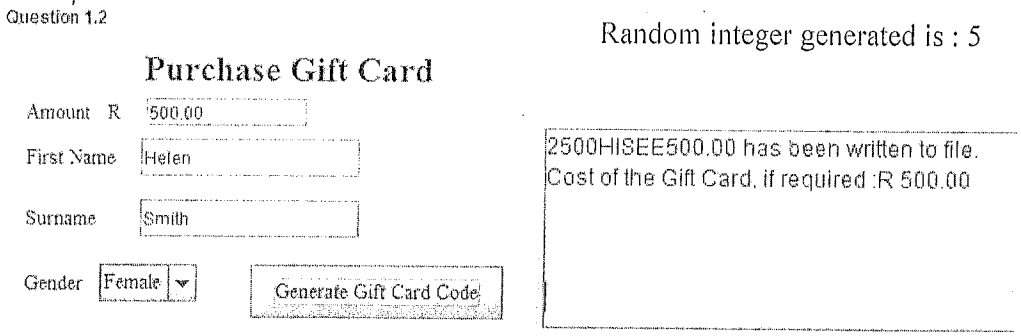
- A number generated as follows:
  - A random integer value in the range 1 to 9 (both inclusive).
  - The above random number must be multiplied with the gift card purchase amount, round off to an integer value.
- Characters as follows:
  - Concatenate the first name and surname.
  - If the customer is a female, convert the concatenated first name and surname to uppercase, else if male, convert to lowercase.
  - Extract every alternate character from the concatenated name in reverse order.
- The gift card purchase amount.

This gift card code must be displayed in the text area, *txaDisplay*, **AND** written to an existing text file named, "GiftCodes.txt".

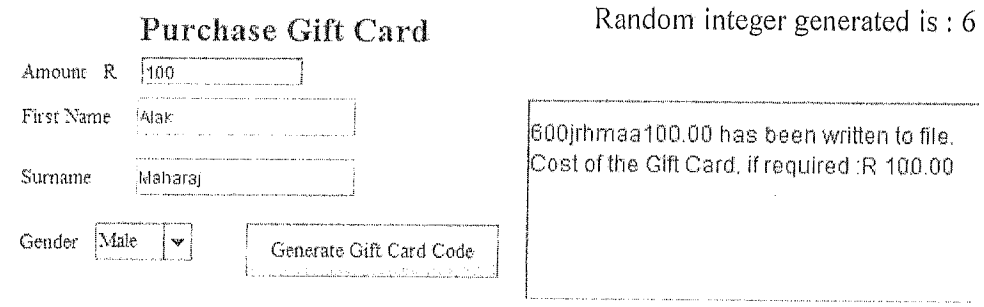
Display the cost of the gift card, in the text area, *txaDisplay*, rounded off to two decimal places.

Example:

Question 1.2



Question 1.2



(19)

1.3 Code the button ***btnFinalPrice***.

1.3.1 CIPU Jewelry is granting a discount to all customers. There are four percentage discounts granted; 2%, 8%, 16% or 32%. The percentage discount granted will be determined as follows:

- The user is to enter a number in the text field, *txfNumber*.
- If this number entered is a **perfect number**
  - Randomly select one percentage discount from the array, ***ArrayDiscount*** provided.

    int ArrayDiscount = {8;16;32};
- If the number is **not a perfect number**, grant a 2% discount only. (11)

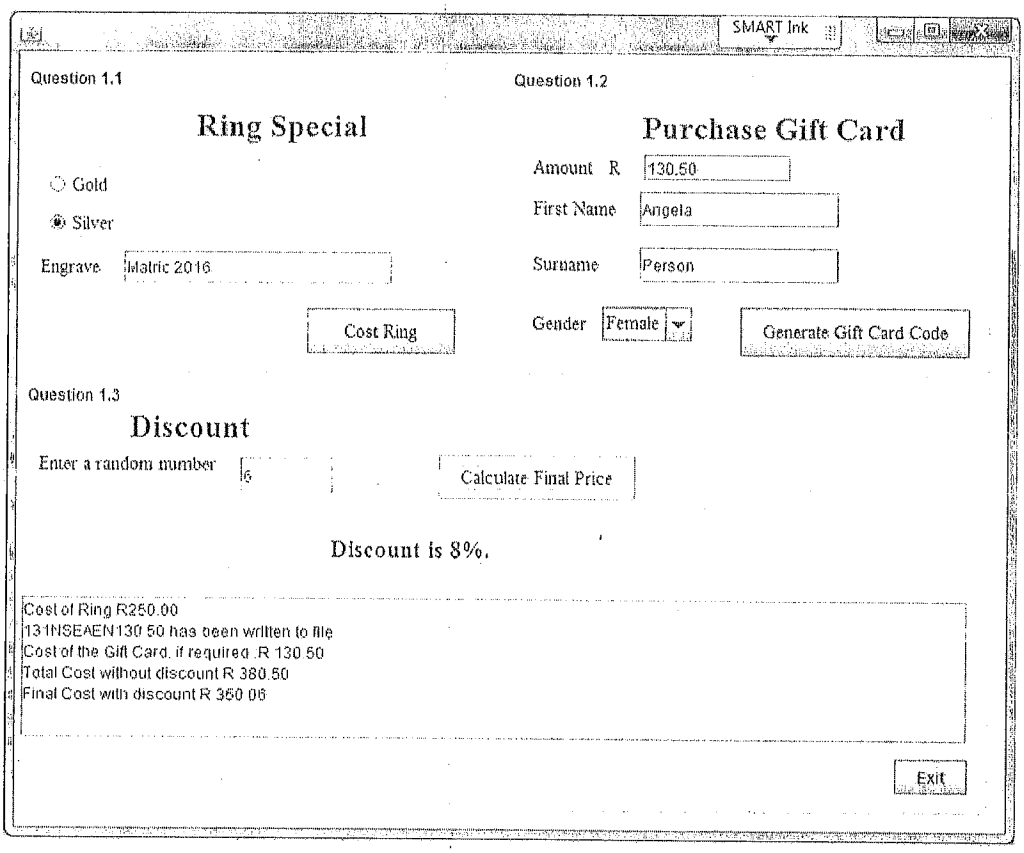
Perfect Number: *If this product the factors, besides the number itself is equal to the number, then that number is regarded as a perfect number.*
Example: Factors of 6 are:
1; 2; 3 and 6. Product of the factors besides the number itself : 1 x 2 x 3 = 6. Therefore 6 is a perfect number.

1.3.2 Display the discount granted in the label, *lblPercentageDiscount*. (1)

1.3.3 Calculate and display in the text area, *txaDisplay*, correct to two decimal places, the:

- The total cost of the purchases.
- The final cost with the percentage discount added. (4)

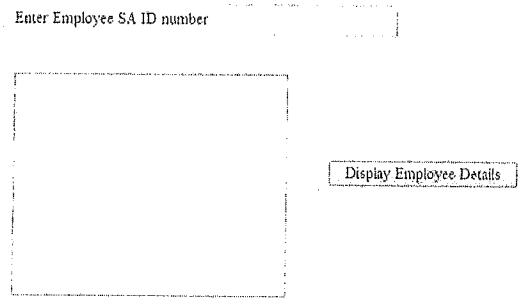
Sample Run:
Random number generated : 1



[57]

QUESTION 2

Open the Java Class, question2Object and the jFrame question2, in the project Question2.



The CEO of CIPU Jewelry's has the following details of all the employees' in a text file named "EmployeeDetails.txt".

2.1 Refer to the class question2Object.java, which consists of four private fields as follows:

ID *: employees South African ID number*
password *: employees password used to log time in and out*
gross *: indicates the employees monthly gross salary (salary before deductions)*
status *: manager, if the employee is a manger of a store.*

The default and parameterized constructors have been coded.

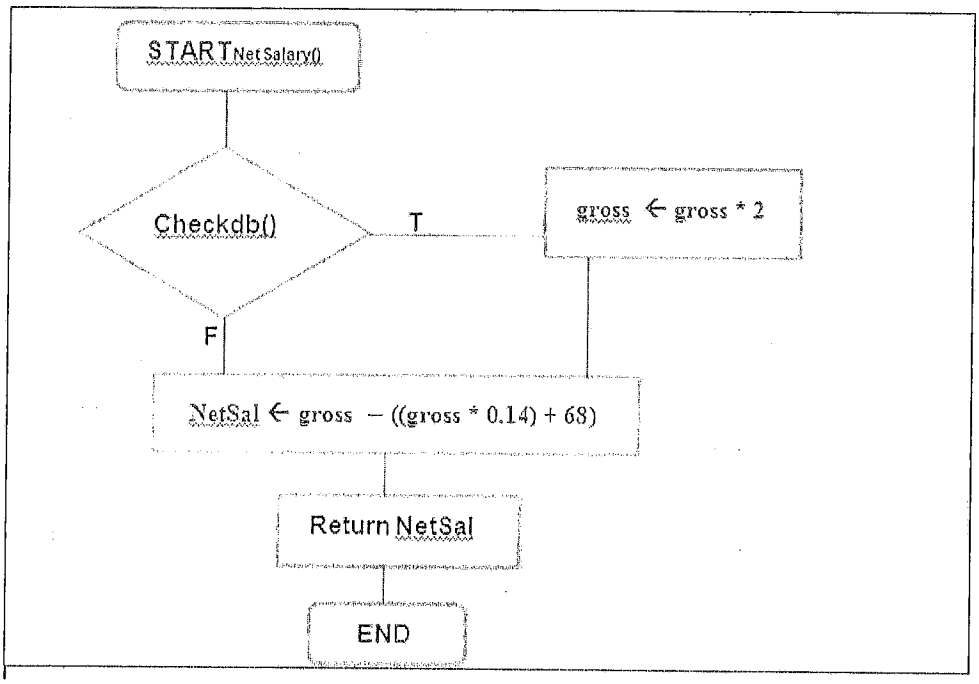
2.1.1 Make use of the pseudo code below to write a Boolean returning method called **Checkdb()**, which will use the computers system clock to extract the current month as well as use the employees South African ID number, to determine if the employee has his/her birthday in the current month.

NB: The first six digits of the South African ID number indicates the citizens date of birth in the format yymmdd.

```
birthday ← false
EmployeeMonth ← 3rd and 4th digit of the ID number
CurrentMonth ←extract current month from the system clock
if (CurrentMonth equals EmployeeMonth)
    birthday ←true
return birthday
```

(8)

2.1.2 Write a method called **NetSalary()**, which will calculate and return the employees net salary (gross salary less deductions), using the flow chart below:



(5)

2.1.3 Write a method called **Strength()** that will return the words; ***Weak, Medium or Strong***, if the employees password meets the following criteria:

Strong	More than or equal to six character AND consist of at least; two digits AND two capital letters AND two ASCII key values from 33 to 47 (both inclusive).
Medium	More than or equal to six characters AND consist of at least; one digit, AND one capital letter AND one ASCII key value from 33 to 47 (both inclusive).
Weak	Less than six characters OR consist of either; zero digits OR zero capital letter OR zero ASCII key values from 33 to 47 (both inclusive).

(20)

2.1.4 Write a **toString()** method, to return a single string in the following format:

Employee ID: employees SA ID number
Password Strength: Weak/Medium/Strong
Gross Salary: employees gross salary
Net Salary: employees net salary
Manager: Yes/NO
<Received Birthday Bonus> (only if the method Checkbd() is true) (8)

2.2 Code the button **btnEmployeeDetails**, in the jFrame quesion2.java, to do the following:

- Extract the search ID number from the text field *txfID*.
- Search for the ID number in the text file "EmployeeDetails.txt". Contents of the text file is as follows:

South African ID number;password;monthly salary^Manager(if employee is a Manager)

```
6403240021082;happy;40500^Manager
7210181813088;%SB897m%;20000
8104190092082;124;23000
6808297810087;*88JNh72$;48000^Manager
....
```

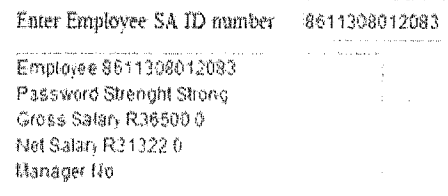
- If the employee ID has been found in the text file, instantiate the object question2Object with the following attributes:
  - ✓ ID number : String
  - ✓ Password : String
  - ✓ Gross salary : Double
  - ✓ Status : String

  NB: The object **q2**, the object of the class question2Object, has already been declared.

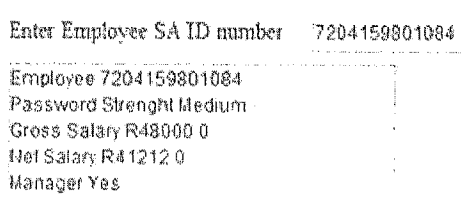
- Display, in the text area, *txaDisplay*, the toString() method from the object class, question2Object. (17)

Sample Run: (Run date : 10 August 2016)

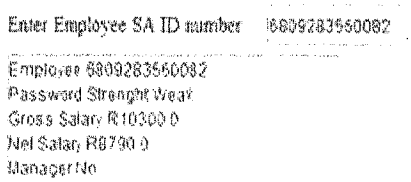
8611308012083;(SmArT)78;36500



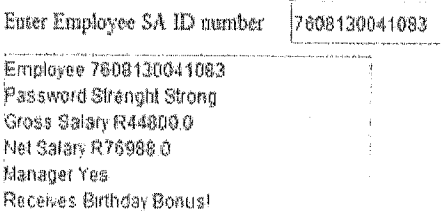
7204159801084;mikE*11;48000^Manager



6809283560082;tammy*;10300



7608130041083;ME13aug$$;44800^Manager



[58]

QUESTION 3

CIPU Jewelry's have two jewelry designers named; Samantha Daniels and Subeko Mabaso. They have one hour time slots, three times a day, available for customers to book a design session.

3.1 Open the jFrame question3_1.java, in the project Question3, which consists of two one-dimensional String arrays and one two-dimensional String array, as follows:

String[ ] arrayDesinger = {"Samantha Daniels","Subeko Mabaso"};

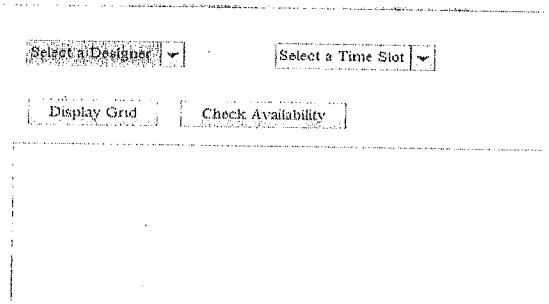
String[ ] arrayTime = {"10:00-11:00","12:00-13:00","14:00-15:00"};

String[ ][ ] arrayBooking = {{"*","0719668925","*",},{"0825698001","*","*"}};

Array arrayDesigner, is populated with the names of the two designers.

Array arrayTime, is populated with the time slots for customers to book.

Array arrayBooking, is populated with either a '*' or a cell number. If populated with a '*', the time slot is available, if populated with a cell number, it is booked by a customer with that cell number.



3.1.1 Code for the button **btnDisplay**, to display the arrays as follows:

	10:00-11:00	12:00-13:00	14:00-15:00
Samantha Daniels	*	0719668925	*
Subeko Mabaso	0825698001	*	*

(8)

3.1.2 Code the button **btnCheckAvailability**, as follows:

- extract the selected designer from the combo box *cmbxDesigner*, and the time slot from the combo box *cmbxTime*.
- Check if the designer and time slot is available.
- If the designer is **NOT** available for the time slot selected, display all the time slots and designers available, e.g. if Samantha Daniels, 12:00-13:00 is selected, the following must be displayed:

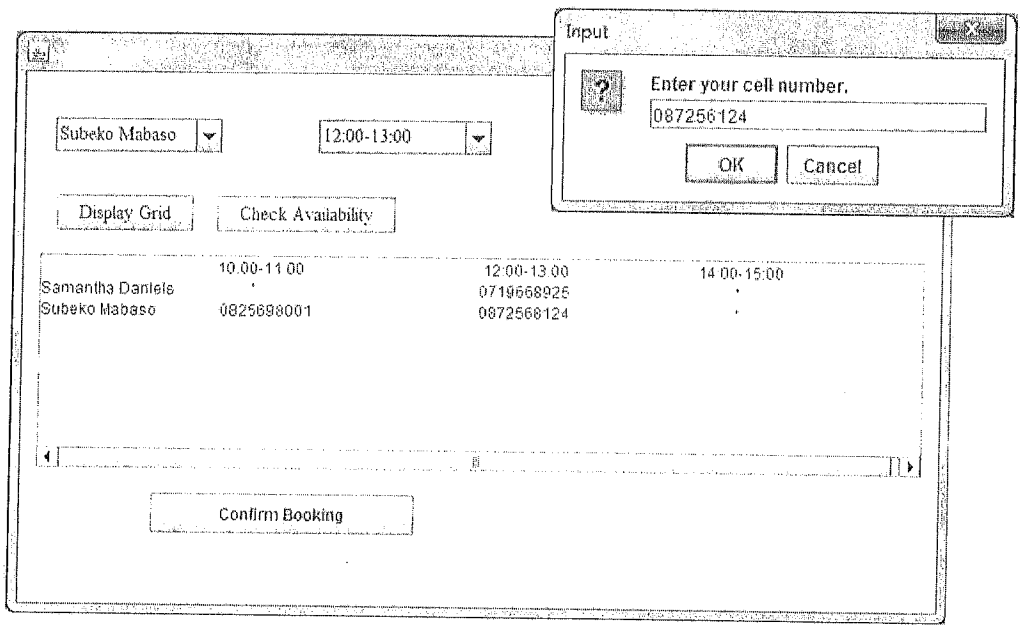
Sorry not available Alternate Time and Designer:	
Samantha Daniels	10:00-11:00
Samantha Daniels	14:00-15:00
Subeko Mabaso	12:00-13:00
Subeko Mabaso	14:00-15:00

- If the designer is available for the selected time slot, instantiate a jButton, with the following specifications and functionality:
  Specifications:

Text on button	Confirm Booking
Panel	jPanel1
Bounds	100,320,200,30

Functionality:
Use an input dialog box to enter the customers cell number. Insert this cell number for the correct designer and time slot in the two-dimensional array, arrayBooking. (18)

Sample Run:
Customer choose Subeko Mabaso for 12:00-13:00. Customer cell number entered in dialog box is 087256124.



3.2 Open the jFrame question3_2.java, in the project Question3.

The maximum time a designer can consult with a client is one hour (60 minutes).

The designers charge a consultation fee as follows:

Consultation time (minutes)	Cost per minute
≤ 5 minutes	Free
> 5 minutes to ≤ 30 minutes	R45.00 per minute
> 30 minutes	R55.00 per minute

Code the button **btnConsultTimeCost**, as follows:

Extract the time in minutes from the spinner, *spnConsultTime*,

- if the time is more than one hour, display in a dialog box, "Consults are a maximum of one hour".
- calculate and display the cost for the consultation in the text area, *txaDisplay*, correct to two decimal places.

NB: You many NOT change the properties of the spinner. (9)

Sample Run:
Time : 25 Cost : R900.00
Time: 35 Cost : R1400.00
Time: 3 Cost : R0.00
Time: 60 Cost : R2775.00

[35]
TOTAL : 150

ASCII Table

Ascii	Char	Ascii	Char	Ascii	Char	Ascii	Char
0	Null	32	Space	64	@	96	`
1	Start of heading	33	!	65	A	97	a
2	Start of text	34	"	66	B	98	b
3	End of text	35	#	67	C	99	c
4	End of transmit	36	$	68	D	100	d
5	Enquiry	37	%	69	E	101	e
6	Acknowledge	38	&	70	F	102	f
7	Audible bell	39	'	71	G	103	g
8	Backspace	40	(	72	H	104	h
9	Horizontal tab	41	)	73	I	105	i
10	Line feed	42	*	74	J	106	j
11	Vertical tab	43	+	75	K	107	k
12	Form feed	44	,	76	L	108	l
13	Carriage return	45	-	77	M	109	m
14	Shift in	46	.	78	N	110	n
15	Shift out	47	/	79	O	111	o
16	Data link escape	48	0	80	P	112	p
17	Device control 1	49	1	81	Q	113	q
18	Device control 2	50	2	82	R	114	r
19	Device control 3	51	3	83	S	115	s
20	Device control 4	52	4	84	T	116	t
21	Neg. acknowledge	53	5	85	U	117	u
22	Synchronous idle	54	6	86	V	118	v
23	End trans. block	55	7	87	W	119	w
24	Cancel	56	8	88	X	120	x
25	End of medium	57	9	89	Y	121	y
26	Substitution	58	:	90	Z	122	z
27	Escape	59	;	91	[	123	{
28	File separator	60	<	92	\	124	|
29	Group separator	61	=	93	]	125	}
30	Record separator	62	>	94	^	126	~
31	Unit separator	63	?	95	_	127	Forward del.

education

Department:
Education
PROVINCE OF KWAZULU-NATAL

INFORMATION TECHNOLOGY

PAPER 1

TRIAL EXAMINATION 2016

MEMO

GRADE 12

DATE: 30 August 2016

MARKS: 150

TIME: 3 hours

EXAMINER : Ms. S BENIMADHO
MODERATOR : Mr. A J PILLAY

Pseudo Code/Algorithm

QUESTION 1 [57 Marks]

1.1.1

Select from both radio buttons ✓
Check if neither metal is selected ✓
Display in a Warning dialog box✓MUST be a warning message

(3)

1.1.2

If radio button gold is selected ✓
Assign 300 to the cost ✓

If radio button silver is selected ✓
Assign 200 to the cost ✓

(4)

1.1.3

Extract the message to be engraved ✓
Find the length of the engraved message ✓
If the length is greater than 15✓
Set the text field to empty ✓

(4)

else / if the length is less than equal to 15 / if the length is less than 16 ✓
Loop the length of the message ✓
Extract the character at the poistio of the loop ✓
If the character is NOT a space ✓
Add one to the number of characters variable ✓

Calculate the cost by multiplying the number of non white characters with 5 and adding on to the cost of the ring ✓

(6)

1.1.4

Display to two decimal places the cost of the ring which will include the basic cost plus the engraved cost
✓display ✓correct to 2 decimal places

(2)

1.2.1

Extract the gift card amount AND type cast ✓ extracting and passing
Extract BOTH name and surname text field ✓
Extract from the combo box ✓

(3)

1.2.2

Randomise a number from 0 to 8 ✓ random 9
Add one to the randomised number ✓ add 1
Random number generated multiplied with the price of gift card ✓ multiply
Round to a whole number the ✓Math.round function
Concatenate name and surname ✓
Convert the concatenated name to uppercase✓ or use another if, or use else
Test if gender equals 'M' ✓
Convert the concatenated name to lowercase ✓

Loop the length -1 of the concatenated name backwards ✓length-1 ✓for loop in reverse b--
Extract the character at position ✓
Decrease loop by one more ✓ or b = b-2 in the if loop

Add price of gift card to code✓

Connect to the text file "GiftCardCodes.txt" that exists ✓Print/buffered ✓true
Write the code to the text file ✓
Close the file ✓
Display the code in the text area ✓
Display the cost of the gift card ✓

(19)

1.3.1

Initiate a variable to to 1 ✓
Extract and parse number ✓
Loop from 1 to the less than the number ✓
MOD (%) the number and the loop variable ✓
Test if the MOD equals 0 ✓
Multiple the factors ✓

If the factors multiplied equals the number as input ✓
Randomise an integer from 0 to 2 inclusive ✓
Extract a discount from the array for the randomised position ✓
else✓ or assign discount to 2 before if
discount is assigned to 2 ✓

(11)

1.3.2

Display the discount variable in the label. ✓

(1)

1.3.3

Display the final cost before the discount ✓
Calculate the final cost after discount ✓ ✓
Display the amount due after discount ✓

(4)

[57]

QUESTION 2 [58 Marks]

2.1.1

Method heading ✓
Assign variable to false ✓
Instantiate the calendar class ✓
Extract the current month from the system clock, add 1 ✓
Extract position 2 and 4 from the ID and parse as integer ✓
If current month = ID month ✓
    Variable assign to true ✓
Return the Boolean variable ✓

(8)

2.1.2

Method heading ✓
If method from 2.1.1 is true ✓
   Gross variable is doubled ✓
Netsalary calculated as given flowchart ✓
Return the netsalasry ✓

(5)

2.1.3

Method heading ✓
Assign to "Weak"or use another if later ✓
Length of the password ✓
Loop the length of the password ✓
        Extract the character at the loop position ✓
        Convert the character to ascii integer value ✓
        If ascii >32 and ascii < 48 ✓ ()✓()
            Increment the variable holding number special characters ✓
        Check if the character is a digit ✓
            Increment the variable holding number digits ✓
        Check if the character is uppercase AND a letter ✓() ✓()
            Increment the variable holding number uppercase letters ✓

If the length >= 6 ✓
        Number special character AND number of digits AND number uppercase letters MUST > 0 ✓all
            Assign to "Medium" ✓
        Number special character AND number of digits AND number uppercase letters MUST >=2✓ all
            Assign to "Strong"; ✓
Return the strength ✓

(20)

2.1.4

Heading of method ✓
Assign variable to "NO" ✓
If the attribute status is "Manager" ✓
        Set the variable to "Yes" ✓
If the method checkdb() from 2.1.3 is true ✓
        Assign a string variable to "Receives Birthday Bonus!"; ✓

    return all attributes and variables as required ✓\n ✓all required data

(8)

2.2

Extract the string ID from the textfield ✓
Link to the external text file called "EmployeeDetails.txt" ✓
Loop until end of file ✓
Extract the line from the text file✓
Split line with ; ✓
Separate the attributes ✓
Check if the last attribute contains a ^ ✓
Find the position of ^✓
Extract from the start to the position on ^ and pass as double for the gross salary ✓substring✓pass
Extract from position of ^+1 to the end for the status ✓
else✓
the last extract is the gross salary ✓
status is set to empty string✓

If the ID entered in the text field match the ID in the file ✓
Instantiate the object class ✓
Call up and display the toString method from the object class ✓

(17)

[58]

QUESTION 3 [35 Marks]

3.1.1
Loop through the arrayTime ✓
Display the array arrayTime ✓

Leave a line✓

Loop the row✓
Display the array arrayDesigner ✓
Loop the columns✓
Display the 2D array arrayBooking ✓
Leave a line ✓

(8)

3.1.2
Extract the INDEX of the combo box Designer ✓
Extract the INDEX of the combo box Time slot ✓

If array arrayBooking at position index -1 NOT equals ✓-1 ✓!EQUALS
Lop the row✓
Loop the column✓
If the array equals *✓
Display array arrayDesigner at row position and array arrayTime at the column position ✓arrayDesigner ✓arrayTime
else✓
Instantiate the button with Text on ✓
Add the button to the panel ✓

Set bound to the button as given ✓
Make the button an event handler ✓

Create a method for the button as an event handler ✓
The array arrayBooking at position index -1 is assigned the value of the input cell number via a dialog box
✓des-1 ✓time-1✓input dialog box

(18)

3.2
Extract the consult time from the spinner ✓
If the time > 60✓
Display a dialog box indicating time of one hour max ✓

If time > 5 AND time <=30 ✓
Subtract 5 from the time and multiply by 45 ✓
If time > 30 AND time <=60 ✓
24 * 45 plus time – 30 multiplied by 55✓ ✓
Display the cost✓

(9)

---

[35]

Java Coding

QUESTION 1 [57 Marks]

1.1.1

```
boolean gold = rbtnGold.isSelected();
boolean silver = rbtnSilver.isSelected();
```

✓ extract both radio buttons

```
if(!gold & !silver) ✓
  JOptionPane.showMessageDialog(null,"Select a type of metal","ERROR",JOptionPane.WARNING_MESSAGE);
```
✓MUST be a warning message

(3)

1.1.2

```
if(gold) ✓
  costRing += 300; ✓

if(silver) ✓ or else
  costRing += 200; ✓
```

(4)

1.1.3

```
engrave = txfEngrave.getText(); ✓
lenEngrave = engrave.length(); ✓
if(lenEngrave > 15) ✓
  txfEngrave.setText(""); ✓
```

(4)

```
else ✓
{
```

```
    for(int a = 0; a<lenEngrave;a++) ✓
    {
      char ch = engrave.charAt(a); ✓
      if(!Character.isWhitespace(ch)) ✓
        numChar++;          ✓
    }
  }
  costRing += (numChar*5); ✓
```

(6)

1.1.4

txaDisplay.append("Cost of Ring R"+d.format(costRing)); ✓display ✓correct to 2 decimal places

(2)

1.2.1

```
costGiftCard = Double.parseDouble(txfAmountGiftCard.getText()); ✓ extracting and passing
String name = txfFirstName.getText();
String surname = txfSurname.getText();   ✓ both
String gender = cmbxGender.getSelectedItem()+""; ✓
```

(3)

1.2.2

```
int rand1 = (int)(Math.random()*9)+1; ✓ random 9   ✓ add 1
int rand = (int)(Math.round(rand1*costGiftCard)); ✓ multiply ✓Math.round function
code = rand+"";
String fullname = name+surname; ✓
fullname = fullname.toUpperCase();✓ or use another if,  or use else
if(gender.equals("Male")) ✓
  fullname = fullname.toLowerCase(); ✓

for(int b = fullname.length()-1;b >-1;b--)✓length-1  ✓for loop in reverse b--
{
  code += fullname.charAt(b); ✓
  b--; ✓ or b = b-2 in the if loop
}
code += d.format(costGiftCard); ✓


  PrintWriter wf = new PrintWriter(new FileWriter("GiftCodes.txt",true)); ✓Print/buffered ✓true
  wf.println(code); ✓
  wf.close(); ✓
  txaDisplay.append("\n"+code+" has been written to file."); ✓
  txaDisplay.append("\nCost of the Gift Card, if required :R "+d.format(costGiftCard)); ✓

}
```

(19)

1.3.1

```
int factors = 1; ✓
int discount = 0;
double finalCost = costRing +costGiftCard;
int number = Integer.parseInt(txfNumber.getText()); ✓

for(int c = 1; c <number ; c++) ✓
{
  if((number%c) == 0) ✓ number%c  ✓ if ==0
    factors = factors * c; ✓
}

if(factors == number) ✓
{
  int pos = (int)(Math.random()*3); ✓
  discount = arrayDiscount[pos]; ✓
}
else✓ or assign discount to 2 before if
  discount = 2; ✓
```

(11)

1.3.2

```
lblPercentageDiscount.setText("Discount is "+discount+"%."); ✓
```

(1)

1.3.3

```
txaDisplay.append("\nTotal Cost without discount R "+d.format(finalCost)); ✓
finalCost = finalCost-(finalCost*discount/100);      ✓multiply ✓ subtract
txaDisplay.append("\nFinal Cost with discount R "+d.format(finalCost)); ✓
```

(4)

[57]

QUESTION 2 [58 Marks]

2.1.1

```
public boolean checkbd() ✓
  {
    boolean birthday = false; ✓
    Calendar now =  Calendar.getInstance(); ✓
    int CurrentMonth = now.get(Calendar.MONTH)+1; ✓
    int EmployeeMonth = Integer.parseInt(ID.substring(2,4)); ✓
    if(CurrentMonth == EmployeeMonth) ✓
      birthday = true; ✓
    return birthday; ✓
  }
```

(8)

2.1.2

```
public double netSal()✓
  {
    double ns =0;

    if(checkbd())✓
    {
      gross = gross*2; ✓
    }
    ns = gross-((gross*0.14)+68); ✓
    return ns; ✓
  }
```

(5)

2.1.3

```
public String strength()✓
  {
    int numUpp=0;
    int numDigit=0;
    int numOther = 0;
    String st = "Weak"; ✓  or another if  or else
    int len = password.length(); ✓

    for(int a = 0; a < len;a++) ✓
    {
      char ch = password.charAt(a); ✓
      int intch = (int)(ch); ✓
      if((intch >32)&(intch <48))        ✓ ()✓()
        numOther++;        ✓
      if(Character.isDigit(ch)) ✓
        numDigit++; ✓
      if(Character.isUpperCase(ch)&(Character.isLetter(ch))) ✓() ✓()
        numUpp++;        ✓
    }

      if(len>=6) ✓
      {
        if((numOther>0)&&(numDigit>0)&&(numUpp>0))        ✓all
          st ="Medium";        ✓
        if((numOther>=2)&&(numDigit>=2)&&(numUpp>=2)) ✓ all
          st = "Strong"; ✓
      }

    return st; ✓
  }
```

(20)

Memo

2.1.4

```
public String toString()✓
{
    String man = "No"; ✓
    if(status.equals("Manager")) ✓
      man = "Yes";✓
    String bd = "";
    if(checkbd())✓
      bd = "Receives Birthday Bonus!"; ✓

    return "Employee "+ID+"\nPassword Strenght "+strength()+"\nGross Salary R"+gross+"\nNet Salary
R"+netSal()+"\nManager "+man+"\n"+bd; ✓\n ✓all required data
  }
```

(8)

2.2

```
private void btnEmployeeDetailsActionPerformed(java.awt.event.ActionEvent evt) {
   double gross = 0;
   String status = "";
   String id = txfID.getText(); ✓
   try
   {
     Scanner file = new Scanner (new FileReader("EmployeeDetails.txt")); ✓
     while(file.hasNext()) ✓
     {
       String line = file.next();✓
       Scanner split = new Scanner(line).useDelimiter(";");✓
       String idf = split.next();
       String pw = split.next();
       String check = split.next();
       if(check.contains("^"))✓
       {
         int pos = check.indexOf("^");✓
         gross = Double.parseDouble(check.substring(0,pos)); ✓substring✓pass
         status = check.substring(pos+1); ✓
       }
       else✓
       {
         gross = Double.parseDouble(check); ✓
         status = "";✓
       }

       if(id.equals((idf))) ✓
       {

        q2 = new question2Object(idf,pw,gross,status); ✓
        txaDisplay.append(q2.toString()); ✓

       }
     }
   }
```

✓ all 5

```
catch(FileNotFoundException e)
{
  JOptionPane.showMessageDialog(null,"File not found!");
}
```

(17)

[58]

QUESTION 3 [35 Marks]

3.1.1

```
private void btnDisplayActionPerformed(java.awt.event.ActionEvent evt) {
for(int c =0;c<3;c++)  ✓
{
 String s = String.format("%-45s%-10s"," ", arrayTime[c]);
   txaDisplay.append(s);  ✓
}
txaDisplay.append("\n");✓
for(int r = 0; r<2;r++)✓
{
   String g = String.format("%-45s", arrayDesinger[r]);
   txaDisplay.append(g);  ✓
   for(int c=0;c<3;c++)✓
   {
      String s = String.format("%-50s", arrayBooking[r][c]);
      txaDisplay.append(s);     ✓
   }
   txaDisplay.append("\n");✓

  }
}
```

(8)

3.1.2

```
private void btnCheckAvailabilityActionPerformed(java.awt.event.ActionEvent evt) {
    des = cmbxDesigner.getSelectedIndex();  ✓
    time = cmbxTime.getSelectedIndex();  ✓


      if(!arrayBooking[des-1][time-1].equals("*"))  ✓-1  ✓ !EQUALS
      {
       txaDisplay.append("Sorry not available Alternate Time and Designer:\n");
        for(int r=0;r<2;r++)✓
       {
         for(int c=0;c<3;c++)✓
         {
            if(arrayBooking[r][c].equals("*"))  ✓

               txaDisplay.append(arrayDesinger[r]+"\t"+arrayTime[c]+"\n");    ✓arrayDesigner ✓arrayTime
         }
       }
     }//if
```

```
        else✓
          {

              JButton jb = new JButton("Confirm Booking");✓
              jPanel1.add(jb); ✓
              jb.setBounds(100,320,200,30); ✓
              jb.addActionListener(new java.awt.event.ActionListener() {
         public void actionPerformed(java.awt.event.ActionEvent evt) {
            BookActionPerformed(evt);
         }
       });

}
  }
```

✓

```
private void BookActionPerformed(java.awt.event.ActionEvent evt) ✓
{
  arrayBooking[des-1][time-1] = JOptionPane.showInputDialog("Enter your cell number."); ✓des-1 ✓time-1✓display
}
```

(18)

3.2

```
    int time = (int)(spnConsultTime.getValue());  ) ✓
    double cost = 0;
    if(time >60) ) ✓
    {
      JOptionPane.showMessageDialog(null,"Consults are maximum of one hour."); ) ✓
    }
    else
    {
     if((time > 5)&(time <=30)) ) ✓
       cost = (time-5)*45;  ✓
     if(time > 30)  ✓
        cost = (25*45)+(time-30)*55; ) ✓ ✓
     txaDisplay.append("Cost for consult is R"+d.format(cost)+"\n"); ✓
    }
```

(9)

---

[35]

TOTAL : 150