

**ICT2612**

( 499985)

October/November 2015

**INTERACTIVE PROGRAMMING**

Duration 2 Hours

50 Marks

**EXAMINERS**

**FIRST**

**MS A THOMAS**

**SECOND**

**MS P VAN DER MERWE**

---

**Closed book examination**

**This examination question paper remains the property of the University of South Africa and may not be removed from the examination venue**

**INSTRUCTIONS**

- **This examination question paper consists of 19 pages plus instructions for completion of a mark reading sheet plus 2 pages for rough work (pp i - ii).**
- **This is a closed book examination.**
- **Answer all the questions on the mark reading sheet provided.**
- **This examination paper consists of 50 questions. Each question counts 1 mark.**
- **Complete the attendance register on the back page, tear it off and hand it to the invigilator**

**GOOD LUCK**

[Please turn over]

**Question 1** (1)

Which one of the following statements is correct?

- 1 Android applications are for mobile devices which can be developed using Eclipse development environment
- 2 Eclipse is exclusively for the development and testing of Android applications
- 3 Android and Eclipse are not open-source technologies
- 4 Android does not have its own software development kit (SDK) but it uses the standard Java SDK

**Question 2** (1)

What is the role of an emulator in developing mobile applications?

- 1 It allows you to design the graphical user interface and code the logic of your mobile applications as well as test these applications for multiple mobile devices
- 2 It helps you to code the logic of your mobile applications
- 3 It allows you to test mobile applications without having access to the mobile devices for which the applications are developed
- 4 It allows you to create graphical user interfaces for mobile applications

**Question 3** (1)

Which one of the following statements explains the contents of `activity_main.xml` file in an Android application project?

- 1 It contains information about the sound and image files used in the application
- 2 It contains information about the default graphical user interface of the application
- 3 It contains code for the class `MainActivity`
- 4 It contains a summary of all the setup information for the application

**Question 4** (1)

Which one of the following is **NOT** a general process associated with developing a mobile application?

- 1 Designing
- 2 Coding
- 3 Compiling
- 4 Inheriting

**Question 5** (1)

Which one of the following is a primitive data type in Java?

- 1 byte
- 2 bool
- 3 Long
- 4 Double

[Please turn over]

**Question 6****(1)**

Which one of the following options is a correct variable declaration?

- 1 float value In \$,
- 2 short postal\_code,
- 3 double static;
- 4 int 1\_score,

**Question 7****(1)**

Which one of the following descriptions is correct about the following lines of code?

```
int i = 90 87, //line 1
double d1 = (double)100 65, //line 2
double d2 = 100, //line 3
char c1 = 10, //line 4
char c2 = '&'; //line 5
```

- 1 Line 5 will give a compiler error
- 2 All lines except lines 2 and 5 will give compiler errors
- 3 Line 1 will give a compiler error
- 4 None of these lines will give compiler errors

Consider the following method definition in a class named Address to answer questions 8 to 10 that follow

```
public static boolean isPostalCode(int postalCode){
}

```

**Question 8****(1)**

What does the keyword `static` in `isPostalCode()` mean?

- 1 It can only be invoked on objects of Address
- 2 You can invoke it without an object of Address
- 3 It can only be invoked in the constructor of Address
- 4 There needs to be at least one object of Address to invoke it

**Question 9****(1)**

What does the keyword `public` in `isPostalCode()` mean?

- 1 It is accessible within Address
- 2 It is accessible within any subclasses of Address
- 3 It is accessible in another class in the same package as Address
- 4 All of the above

[Please turn over]

**Question 10**

(1)

The method `isPostalCode()` is meant to check if a given `int` (`postalCode`) is a postal code. A valid postal code is a 4 digit code. If `postalCode` contains a valid postal code, the method returns `true` or else it returns `false`. Which one of the following options correctly codes these requirements?

- 1 `if(String.valueOf(postalCode).length() == 4)`  
    `return true,`  
    `else`  
        `return false,`
- 2 `if(postalCode.length() == 4)`  
    `return true,`  
    `else`  
        `return false,`
- 3 `if(postalCode < 1000)`  
    `return true,`  
    `else`  
        `return false,`
- 4 `if(postalCode > 0 && (postalCode < 1000))`  
    `return true,`  
    `else`  
        `return false,`

**Question 11**

(1)

An employment agency is asked to shortlist candidates for an interview that satisfies **either one** of the following conditions:

- the candidate is younger than 25 and obtained an average of at least 60% for his/her undergraduate degree/diploma
- the candidate is a Unisa graduate who has finished his/her undergraduate degree/diploma in no more than three years

Which one of the following expressions correctly selects the candidates? Assume that all the variables are declared and initialised correctly.

- 1 `((age < 25)&&(average >=60)) && ((unisaGrad == true) && (numYears <=3))`
- 2 `((age < 25)&&(average >=60)) || ((unisaGrad == true) && (numYears <=3))`
- 3 `((age < 25)|| (average >=60)) || ((unisaGrad == true) || (numYears <=3))`
- 4 `((age < 25)&&(average >=60)) || ((unisaGrad == true) || (numYears <=3))`

[Please turn over]

**Question 12****(1)**

Which one of the following options is correct regarding these three statements?

```
int i = 10/3, //line 1
double j = 10 0/5 0, //line 2
double k = 10/5, //line 3
```

- 1 Lines 2 and 3 compile without error but Line 1 will not compile
- 2 The result of the expressions are  $i = 3$ ,  $j = 2$ ,  $k = 2$
- 3 The result of the expressions are  $i = 3$ ,  $j = 2 0$ ,  $k = 2 0$
- 4 The result of the expressions are  $i = 3$ ,  $j = 2 0$ ,  $k = 2$

**Question 13****(1)**

Which one of the following statements is correct regarding exception handling in Java?

- 1 When correctly used, exception handling can be used to deal with both syntax and logical errors
- 2 It is always optional to handle exceptions
- 3 Exception handling can be used in such a way that a program quits without crashing
- 4 Some of the relevant keywords for exception handling are `exception`, `try` and `throw`

**Question 14****(1)**

Choose the correct data types for the variables in the code below

```
temperatureInCelcius = 30 00,
celciusToFahrenheit = 9 0/5,
temperatureInFahrenheit = temperatureInCelcius * celciusToFahrenheit + 32,
```

- 1 `temperatureInCelcius` and `temperatureInFahrenheit` should be double and `celciusToFahrenheit` should be int
- 2 It can be either float or double
- 3 All should be float
- 4 All should be double

Questions 15 to 16 are based on the code below:

```
int result = 0,  
int num = 4,  
for(int i = num, i > 0, i--)  
    result += i,
```

**Question 15**

(1)

What is the value stored in `result` after the execution of these lines of code?

- 1 0
- 2 1
- 3 10
- 4 16

**Question 16**

(1)

Which one of the following options is essentially the same as the given code?

```
1 int result = 0,  
  int num = 4,  
  for(int i = 0, i >= num, i++){  
    result += i,  
  }
```

```
2 int result = 0,  
  int num = 4,  
  int i = 0,  
  while( i <= num) {  
    result += i,  
    i = i + 1,  
  }
```

```
3 int result = 0,  
  int num = 4,  
  do{  
    result += num,  
  } while(num > 0),
```

```
4 int result = 0,  
  int num = 4,  
  do{  
    result += --num,  
  } while(num != 0),
```

**Question 17****(1)**

What is the value stored in `s2` after the execution of the following lines of code?

```
String s1 = "Good Luck!",  
String s2 = new String(),  
for(int i = s1.length()-1, i >=0, i--){  
    s2 = s2.concat(String.valueOf(s1.charAt(i))),  
}
```

- 1 !Luck Good
- 2 'Good Luck
- 3 'dooG kcuL
- 4 'kcuL dooG

**Question 18****(1)**

What is the value stored in `province` after the execution of the following lines of code?

```
String input = "WC",  
String province,  
  
switch(input) {  
    case "L" province = "Limpopo",  
    case "NC" province = "Northern Cape",  
    case "GP" province = "Gauteng",  
    case "WC" province = "Western Cape",  
    default province = "North West",  
}
```

1. Western Cape
- 2 Western Cape North West
- 3 North West
- 4 It will generate a compiler error since a `String` cannot be used as the expression in a `switch` statement

**Question 19****(1)**

Which one of the following options is correct with respect to `Strings`?

- 1 Both the method `equals()` and the operator `==` can be used to check if the text stored in two `String` objects are the same
- 2 `Strings` are mutable objects
- 3 You can use either the attribute `length` or the method `length()` to obtain the length of a string stored in a `String` object
- 4 `String[] stringArray = {"", ""}`, is a valid statement

**Question 20**

(1)

Which one of the following statements is correct regarding classes and/or objects?

- 1 A class is a blueprint of a concept and an object is an instantiation of a class
- 2 Invoking the constructor of a class is optional for creating its objects
- 3 Methods in classes cannot have local variables
- 4 In Java, a class may not have multiple constructors

Study the code and the description given below to answer questions 21 to 30 that follow

```
class Inventory{  
  
    protected String[] productName,  
    protected int [] inStock,  
    protected double[] productPrice,  
  
    int totalProducts,  
  
    public Inventory(){  
        // (i)  
    }  
  
    public void addProduct(String name, int number, double price){  
        // (ii)  
    }  
  
    public void updateQuantity(String name, int number){  
        // (iii)  
    }  
  
}
```

The class `Inventory` represents an inventory list with product names, number of items in stock and the prices of products in three different arrays `productName`, `inStock` and `productPrice` respectively. For a product stored in the index `i` of `productName`, the number of items in stock and its price can be found in index `i` of `inStock` and `productPrice` respectively. A maximum of 100 products are allowed in an inventory. However the number of products in the inventory at a given time is indicated in `totalProducts`.

**Question 21**

(1)

Which one of the following options indicates the class variables of `Inventory`?

- 1 `productName`, `inStock`, `productPrice`
- 2 `totalProducts`
- 3 `name`, `number`, `price`
- 4 No class variables declared

[Please turn over]



**Question 22****(1)**

Which one of the following options indicates the instance variables of Inventory?

- 1 productName, inStock, productPrice
- 2 totalproducts
- 3 name, number, price
- 4 No instance variables declared

**Question 23****(1)**

Which one of the following options indicates the local variables in Inventory?

- 1 productName, inStock, productPrice
- 2 totalproducts
- 3 name, number, price
- 4 No local variables declared

**Question 24****(1)**

Which one of the following options is the correct implementation of the Inventory constructor (i) that satisfies the requirements stated for Inventory?

- 1 

```
public void Inventory(){
    totalProducts ++,
}
```
- 2 

```
public Inventory(){
    totalProducts = 0,
}
```
- 3 

```
public Inventory(){
    String[] productName = new String[100],
    int[] inStock = new int[100],
    double[] productPrice = new double[100],
    totalProducts = 0,
}
```
- 4 

```
public Inventory(){
    productName = new String[100],
    inStock = new int[100],
    productPrice = new double[100],
    totalProducts = 0,
}
```

**Question 25**

(1)

Based on the code for Inventory, how can you count the number of instances of this class?

- 1 It is not possible to count the number of instances
- 2 totalProducts indicate the number of instances
- 3 Length of the array productName can be used to get the number of instances
- 4 Length of any of the three arrays can be used to get the number of instances

**Question 26**

(1)

Which one of the options is correct?

- 1 Inventory i = new Inventory(100),
- 2 Inventory i = new Inventory(),
- 3 Inventory i(100),
- 4 Inventory i(),

**Question 27**

(1)

The method addProduct() (ii) is meant to add the values in name, number and price to the respective arrays and increment the total products in the inventory list Which one of the following correctly implements this logic?

- 1 public void addProduct(String name, int number, double price){  
    int i = productName.length,  
    productName[i-1] = name,  
    inStock[i-1] = number,  
    productPrice[i-1] = price,  
}
- 2 public void addProduct(String name, int number, double price){  
    int i = productName.length;  
    productName[i+1] = name,  
    inStock[i+1] = number;  
    productPrice[i+1] = price,  
}
- 3 public void addProduct(String name, int number, double price){  
    productName[totalProducts] = name,  
    inStock[totalProducts] = number,  
    productPrice[totalProducts] = price,  
    totalProducts++,  
}
- 4 public void addProduct(String name, int number, double price){  
    productName[totalProducts - 1] = name,  
    inStock[totalProducts - 1] = number,  
    productPrice[totalProducts - 1] = price,  
    totalProducts++,  
}

**Question 28****(1)**

Which one of the following is the correct reason for a possible runtime error in `addProduct()`?

- 1 Adding a product that already exists in the `productName` array
- 2 Adding a product name in `productName` but not adding appropriate values in `inStock` and `productPrice` arrays
- 3 Adding an empty `String` object as the product name in `productName`
- 4 Adding a product name in `productName` in index 100

**Question 29****(1)**

The method `updateQuantity()` (iii) should update the number of items in stock if the name of the product already exists in the inventory. Which one of the following correctly implements this logic?

- 1 

```
public void updateQuantity(String name, int number){
    for(int i = 0, i < totalProducts, i++){
        if(name equals(productName[i]))
            inStock[i] = number,
    }
}
```
- 2 

```
public void updateQuantity(String name, int number){
    for(int i = 0, i < totalProducts length, i++){
        if(name equals(productName[i]))
            inStock[i] = number,
    }
}
```
- 3 

```
public void updateQuantity(String name, int number){
    for(int i = 0, i < productName length, i++){
        if(name == productName[i])
            inStock[i] = number,
    }
}
```
- 4 

```
public void updateQuantity(String name, int number){
    for(int i = 0, i < productName length, i++){
        inStock[i] = number,
    }
}
```

**Question 30**

(1)

Assume you want a different inventory list for perishable items so that their sell by dates (in addition to the name, no of items in stock and price) can also be stored Which one of the following options is the best design choice for this requirement?

- 1 Add a fourth array that can store dates in Inventory
- 2 The class Inventory inherits from the FoodInventory class where an array of dates is declared in FoodInventory
- 3 class FoodInventory extends Inventory where an array of dates is declared in FoodInventory
- 4 class FoodInventory extends Inventory and both these classes include an array of dates each

**Question 31**

(1)

Which one of the following options is the correct declaration and/or initialization of an array?

- 1 `String[] provinces = new String[],`
- 2 `String[] provinces,  
provinces = new String[];`
- 3 `String[] provinces = new String[9],  
provinces = {"Eastern Cape", "Free State", "Gauteng", "KwaZulu Natal",  
"Limpopo", "Mpumalanga", "Northern Cape", "Western Cape", "North West"},`
- 4 `String[] provinces = {"Eastern Cape", "Free State", "Gauteng", "KwaZulu  
Natal", "Limpopo", "Mpumalanga", "Northern Cape", "Western Cape", "North  
West"},`

**Question 32**

(1)

Which one of the following is **incorrect** regarding arrays?

- 1 The size of an array can be changed after declaring and/or initializing the array
- 2 If the size of the array is 10, the index of its last element is 9
- 3 Arrays can store primitive data types or objects
- 4 An array can only hold data of the same type

**Question 33**

(1)

What is the order in which elements are stored in `numbers` after the execution of the following lines of code?

```
int[] numbers = new int[3],
numbers[0] = 45,
numbers[1] = 3,
numbers[2] = 25,
Arrays.sort(numbers),
```

- 1 45 25 3
- 2 3 25 45
- 3 45 3 25
- 4 25 3 45

**Question 34**

(1)

What is the value stored in `result` after the execution of the following lines of code?

```
int[] numbers = new int[3],
numbers[0] = 45,
numbers[1] = 3,
numbers[2] = 25,
Arrays.sort(numbers);
int result = Arrays.binarySearch(numbers, 3),
```

- 1 0
- 2 1
- 3 2
- 4 3

**Question 35**

(1)

Which one of the following statements is correct regarding the following lines of code?

```
int speed = 40;
boolean isSpeeding = (speed > 60)? true false,
String result = (isSpeeding == false)? "No speeding fine" "You
have a speeding fine",
```

- 1 The result of a ternary operator cannot be boolean
- 2 There cannot be two statements consecutively using ternary operators
- 3 When the lines of code are executed the string stored in `result` is No speeding fine
- 4 When the lines of code are executed the string stored in `result` is false No speeding fine

**Question 36**

(1)

Which one of the following statements is correct regarding exceptions in Java?

- 1 It refers to errors which include syntax, logical or runtime errors
- 2 Handling of exceptions is always optional in Java although it is a good practice to handle them in programs
- 3 An example of a checked exception is array index out of bounds exception
- 4 An example of an unchecked exception is the null pointer exception

**Question 37**

(1)

Which one of the following statements is correct regarding exceptions in Java?

- 1 Exception handling can only be done using the try, catch constructs
- 2 If you do not want to handle an exception in a method where an exception may arise you can delegate the responsibility of handling the exception to its calling method using throws
- 3 One can include statements in a try block without a catch block
- 4 One can have a catch block without a try block

**Question 38**

(1)

Which one of the following will compile correctly?

- 1 

```
public int divide(int num1, int num2){
    int result = 0,
    try{
        result = num1/num2,
    }
    catch(){
    }
    return result,
}
```
- 2 

```
public int divide(int num1, int num2){
    int result = 0,
    try {
        result = num1/num2,
    }
    catch(Exception e){
    }
    return result;
}
```
- 3 

```
public int divide(int num1, int num2){
    int result = 0;
    try(Exception){
        result = num1/num2,
    }
    catch(){
    }
    return result,
}
```

[Please turn over]

```

4 public static int divide(int num1, int num2){
    int result = 0,
    try{
        result = num1/num2,
    }
    catch(Exception e),
    return result,
}

```

**Question 39**

(1)

Which one of the following options is correct regarding `isPostalCode()` given below?

```

public static boolean isPostalCode(String postalCode) throws Exception{
    boolean result = false,
    int length = postalCode length();
    if(length == 4){
        Integer parseInt(postalCode),
        result = true,
    }
    return result,
}

```

- 1 The `throws Exception` in the method ensures that any runtime error from this method will not let the program crash (or end abruptly)
- 2 There is no possibility of errors arising from this method and thus the exception handling is unnecessary
- 3 The compiler forces the method that calls `isPostalCode()` to handle the exception
- 4 This method generates a compiler error because the result of the expression `Integer parseInt(postalCode)`, is not assigned to an `int`

Questions 40 – 43 refer to the class `Module` given below

```

class Module{
    public Module(){
        this moduleName = "Unknown",
        this moduleLeader = "Unknown",
        this numOfStudents = -1,
    }
    public Module(String name, String leader){
        this moduleName = name,
        this moduleLeader = leader,
        this numOfStudents = -1,
    }
    public Module(String name, String leader, int num){
        this moduleName = name,
        this moduleLeader = leader,
        this numOfStudents = num,
    }
    private String moduleName,
    private String moduleLeader,
    private int numOfStudents,
}

```

[Please turn over]

**Question 40**

(1)

Which one of the following correctly describes the class `Module`?

- 1 It demonstrates inheritance
- 2 It demonstrates method overriding
- 3 It demonstrates method overloading
- 4 `Module` cannot be instantiated

**Question 41**

(1)

Which one of the following options will not render an error?

- 1 `Module m1 = new Module ("ICT2612", 600),`
- 2 `Module m1 = new Module("R van der Merwe", "ICT2612"),`
- 3 `Module m1 = new Module(-1),`
- 4 `Module m1 = new Module(ICT2612, R van der Merwe, 600),`

**Question 42**

(1)

Which one of the following options is correct regarding the statement `Module m1 = new Module(),?`

- 1 It creates an object but the values of `moduleName`, `moduleLeader` and `numOfStudents` cannot be predicted
- 2 It creates an object with default values "" for `moduleName` and `moduleLeader` and 0 for `numOfStudents`
- 3 It invokes the default constructor provided by the compiler
- 4 It creates an object with values `Unknown` for `moduleName` and `moduleLeader` and 1 for `numOfStudents`

**Question 43**

(1)

Which one of the following correctly describes the scope of the variables `moduleName`, `moduleLeader` and `numOfStudents` in `Module`?

- 1 It is generally accessible within all the methods of `Module`
- 2 It can be accessed using an object of `Module` outside the class Example `m1 moduleName = "COS2614"`, where `m1` is an instance of `Module`
- 3 It can be accessed directly by the methods of a subclass of `Module`
- 4 It can be accessed directly by the methods of any class that is in the same Android project as `Module`

**Question 44**

(1)

Which one of the following is correct regarding inheritance in general?

- 1 Inheritance introduces code duplication
- 2 All instance variables of a superclass can be inherited and accessed directly in its subclasses
- 3 Inheritance can be seen as a relationship between two classes
- 4 The parent class is meant to be more specialized than its child class

[Please turn over]



**Question 45****(1)**

Which one of the following keywords is used in code relating to inheritance in Java?

- 1 `extend`
- 2 `super`
- 3 `inherits`
- 4 `exception`

**Question 46****(1)**

Which one of the following examples demonstrates the best design choice for implementing inheritance?

- 1 When designing an application to manage personal assets, a class named `Asset` can be a superclass and a type of asset savings can be designed as a subclass of `Asset`, namely `Savings`
- 2 When designing an application to handle shapes, a class named `Circle` can be the superclass and a class name `Shape` can be its subclass
- 3 When designing an application for a bookstore, a class named `EducationalBook` can be a superclass and a class name `Book` can be made its subclass
- 4 When designing an application to manage bank accounts, transactions in a bank account can be modeled using a superclass namely `Deposit` and its subclass namely `Transaction`

**Question 47****(1)**

Select the option that is **NOT** directly related to writing to files in Java

- 1 `FileNotFoundException`
- 2 `IOException`
- 3 `BufferedWriter`
- 4 `writeToFile`

**Question 48**

(1)

What will be stored in result after executing these lines of code for the contents of file named modules.txt that is placed after the code?

```
String result = "",
BufferedReader br = new BufferedReader(new FileReader(new
File("modules.txt"))),
String inString = "",
StringBuilder sb = new StringBuilder(),
while ((inString = br.readLine()) != null){
    sb.append(inString),
}
br.close(),
result = sb.toString(),
```

modules - Notepad

```
File Edit Format View Help
COS1501 . NQF (5)
COS2614 . NQF (6)
COS3711 . NQF (7)
```

```
1 COS1501 NQF (5)
   COS2614 NQF (6)
   COS3711 : NQF (7)
2 COS1501:NQF(5)COS2614 NQF(6)COS3711 NQF(7)
3 COS1501 NQF (5)COS2614 . NQF (6)COS3711 . NQF (7)
4 COS1501NQF(5)COS2614NQF(6)COS3711NQF(7)
```

**Question 49**

(1)

Which one of the following definitions of the method readFromFile() will compile successfully?

```
1 public static String readFromFile() throws FileNotFoundException,
   IOException{
   String result = "";
   BufferedReader br = new BufferedReader(new FileReader(new
       File("modules.txt"))),
   String inString = "",
   StringBuilder sb = new StringBuilder(),
   while ((inString = br.readLine()) != null){
       sb.append(inString),
   }
   br.close();
   result = sb.toString();
   return result,
}
```

```

2 public static String readFromFile() throws FileNotFoundException{
    String result = "",
    BufferedReader br = new BufferedReader(new FileReader(new
        File("modules txt"))),
    String inString = "",
    StringBuilder sb = new StringBuilder(),
    while ((inString = br readLine()) != null){
        sb append(inString),
    br close();
    result = sb toString(),
    return result,
}

```

```

3 public static String readFromFile(){
    String result = "",
    BufferedReader br = new BufferedReader(new FileReader(new
        File("modules txt"))),
    String inString = "",
    StringBuilder sb = new StringBuilder(),
    while ((inString = br readLine()) != null){
        sb append(inString);
    br close(),
    result = sb toString(),
    return result,
}

```

```

4 public static String readFromFile() {
    String result = "",
    try{
        BufferedReader br = new BufferedReader(new FileReader(new
            File("modules txt"))),
        String inString = "",
        StringBuilder sb = new StringBuilder(),
        while ((inString = br readLine()) != null){
            sb append(inString),
        }
        br close(),
    } catch(IOException e1, FileNotFoundException e2){}
    result = sb toString(),
    return result,
}

```

**Question 50****(1)**

Which one of the following Java topics is **NOT** covered in the syllabus of ICT2612?

- 1 handling of exceptions
- 2 class definitions
- 3 manipulation of arrays
- 4 implementing multithreading

Rough work (do not remove from examination paper)



Rough work (do not remove from examination paper)

## MARK READING SHEET INSTRUCTIONS

Your mark reading sheet is marked by computer and should therefore be filled in thoroughly and correctly

**USE ONLY AN HB PENCIL TO COMPLETE YOUR MARK READING SHEET**

*PLEASE DO NOT FOLD OR DAMAGE YOUR MARK READING SHEET*

Consult the illustration of a mark reading sheet on the reverse of this page and follow the instructions step by step when working on your sheet

Instruction numbers ① to ⑩ refer to spaces on your mark reading sheet which you should fill in as follows

- ① Write your paper code in these eight squares, for instance

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| P | S | Y | 1 | 0 | 0 | - | X |
|---|---|---|---|---|---|---|---|

- ② The paper number pertains only to first-level courses consisting of two papers

WRITE 

|   |   |
|---|---|
| 0 | 1 |
|---|---|

 for the first paper and 

|   |   |
|---|---|
| 0 | 2 |
|---|---|

 for the second. If only one paper, then leave blank

- ③ Fill in your initials and surname
- ④ Fill in the date of the examination
- ⑤ Fill in the name of the examination centre
- ⑥ WRITE the digits of your student number HORIZONTALLY (from left to right). Begin by filling in the first digit of your student number in the first square on the left, then fill in the other digits, each one in a separate square
- ⑦ In each vertical column mark the digit that corresponds to the digit in your student number as follows [-]
- ⑧ WRITE your unique paper number HORIZONTALLY  
NB Your unique paper number appears at the top of your examination paper and consists only of digits (e.g. 403326)
- ⑨ In each vertical column mark the digit that corresponds to the digit number in your unique paper number as follows [-]
- ⑩ Question numbers 1 to 140 indicate corresponding question numbers in your examination paper. The five spaces with digits 1 to 5 next to each question number indicate an alternative answer to each question. The spaces of which the number correspond to the answer you have chosen for each question and should be marked as follows [-].
- ◆ For official use by the invigilator. Do not fill in any information here