

New Specification



Rewarding Learning
ADVANCED SUBSIDIARY (AS)
General Certificate of Education
2017

Centre Number
[][][][][]

Candidate Number
[][][][]

Software Systems Development

Unit AS1:

Introduction to Object Oriented Development



[SDV11]

MONDAY 15 MAY, AFTERNOON

TIME

2 hours.

INSTRUCTIONS TO CANDIDATES

Write your Centre Number and Candidate Number in the spaces provided at the top of this page.
Write your answers in the spaces provided in this question paper.
Answer **all six** questions.

INFORMATION FOR CANDIDATES

The total mark for this paper is 100.
Figures in brackets printed down the right-hand side of pages indicate the marks awarded to each question or part question.

ADVICE TO CANDIDATES

You are advised to take account of the marks for each part question in allocating the available examination time.

For Examiner's use only			
Question	Marks available	Marks	Remark
1	9		
2	23		
3	23		
4	14		
5	24		
6	7		
Total	100		

1 Complete the following statements about an object-oriented programming environment by inserting the appropriate words or phrases from the list given below.

overloading interfaces inheritance classes polymorphism
signatures overriding implements methods late binding
new Object early binding object instantiation

- (i) _____ is the creation of an object to the design of a class.
- (ii) _____ is the base building block of an object-oriented system and all _____ are derived from it.
- (iii) _____ occurs where two or more methods have the same name but different _____, even if their return types differ.
- (iv) Customising a super / base method within a derived/sub class is known as _____.
- (v) Multiple inheritance is implemented in C#/Java through the use of _____.
- (vi) A primary concept of object-oriented programming is _____. It allows sub/derived class methods to be invoked through a super / base class reference during run-time. This is enabled through _____.

[9]

Examiner Only	
Marks	Remark

- 2 An artist runs a restoration service where donated items or those retrieved from recycling, are restored or turned into alternative reusable items. These are then offered for sale through upmarket retailers. Restorations fall into three main categories, A – Lamps, B – Tables and C – Sundry.

The production cost is the total of the labour cost, materials cost and specialism cost. The labour cost is the number of hours multiplied by the rate of £20 per hour. Specialism costs occur for outsourced work when required.

The sale price is determined by increasing the production cost by 40%.

- (a) Complete the design of a class called **Job** shown below.

Ensure the creation of:

- A constant variable to hold the value 20 for the hourly rate;
- **Get** and **Set** (Properties / Methods) for the field **noHours** only;
- **Methods** to determine appropriate costs.

```
class Job{
    private int      jobNo;
    private String   description;
    private char     category;
    private DateTime jobDate;
    private DateTime saleDate;
    private int      noHours;
    private double   materialCost; // examples – paint, material
    private double   specialismCost;

    // constant variable to hold the value 20 for the hourlyRate.
```

[2]

Examiner Only	
Marks	Remark

BLANK PAGE
(Questions continue overleaf)

3 Sorting is a common activity conducted on data in information processing.

(a) (i) Name and describe a simple sorting method.

[5]

Examiner Only	
Marks	Remark

(b) C# and Java provide a SORT class to facilitate sorting requirements.

Complete the sentence below.

(i) To allow use of the SORT facility a class header must implement the _____ called _____. [2]

(ii) For the following class Product, a list of product details is required where the product category is in alphabetical order and the price within each category is in ascending order.

C#

```
class Product : IComparable{
    private int      productNo;
    private String   description;
    private String   category;
    private int      noOfProduct;
    private double   price;
```

java

```
class Product implements Comparable{
    private int      productNo;
    private String   description;
    private String   category;
    private int      noOfProduct;
    private double   price;
```

Write the method that will accomplish this requirement by comparing two product objects and return an integer that is:

- negative if the first object is 'less than' the second object;
- zero if the objects are equal;
- positive if the first object is 'greater than' the second object.

*Note that the class **String** facilitates a SORT with a **compareTo** method that returns an integer to indicate the alphabetical order of the two String objects being compared.*

Hint: the price only needs to be considered if the two product items have the same category.

Examiner Only	
Marks	Remark

BLANK PAGE

- (c) Below is a section of code to deal with a file of stock objects. Choosing either the C# or Java example explain what you understand by the emboldened terms.

Section of C# code

```
int size=0;
Stream strm;
try
{
strm = File.OpenRead("Stock.dat");
BinaryFormatter bf = new BinaryFormatter( );

try
{
    while (strm.Position < strm.Length)
    {
        arrayStock[size] = (Stock)bf.Deserialize(strm);
        size++;
    }
    strm.Close();
}
}
catch
```

Section of Java code

```
int size = 0;
try{
FileInputStream strm = new FileInputStream("Stock.dat");
ObjectInputStream ois = new ObjectInputStream(strm);

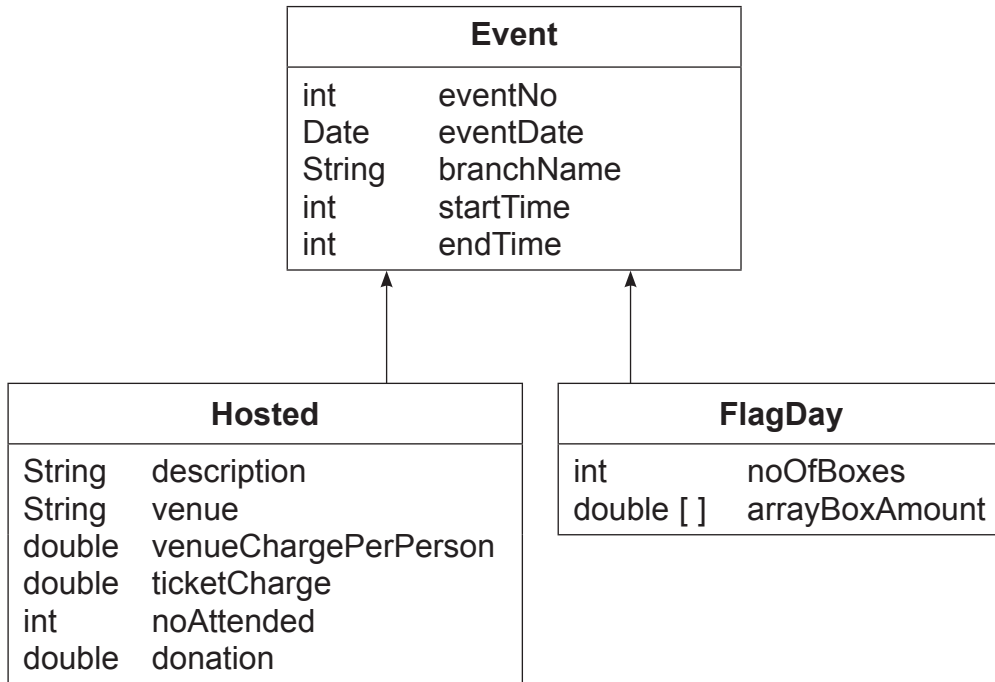
try
{
    while (strm.Position < strm.Length)
    {
        arrayStock[size] = (Stock)ois.readObject();
        size++;
    }
    strm.Close();
}
}
catch(
```

Examiner Only	
Marks	Remark

- 5 A charity’s fund-raising section supports a variety of events run by volunteer branches throughout the country. Common events are flag days, breakfasts, lunches, dinner dances and sporting events.

The charity holds information about each event for the purposes of advertising and the tracking of income. The volunteer branch deals with the planning details.

Inheritance diagram for the charity fund-raising section.



The header for the class **Event** has been defined as follows:

abstract class Event

- (a) Explain the term in bold indicating why it has been applied to the class Event.

[2]

Examiner Only	
Marks	Remark

Assuming the class Event has been designed with the following elements

- field definitions;
- default and field/parameterised constructors;
- GET and SET (Properties/Methods);
- toString() method.

(b) Write the code for a method **HostedIncome** that will return the income generated from a **Hosted** Event.

Income is calculated as profit on a ticket, multiplied by noAttended, plus donations. The profit on a ticket is calculated as the ticketCharge minus the venueChargePerPerson.

[4]

Examiner Only	
Marks	Remark

Permission to reproduce all copyright material has been applied for.
In some cases, efforts to contact copyright holders may have been unsuccessful and CCEA
will be happy to rectify any omissions of acknowledgement in future if notified.