Rewarding Learning

ADVANCED SUBSIDIARY (AS)
General Certificate of Education
2016

Centre Number

Candidate Number

Software Systems Development

Unit AS1:

Introduction to Object Oriented Development

A1S11

[A1S11]
WEDNESDAY 25 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	6		
2	16		
3	27		
4	13		
5	24		
6	14		
Total	100		

1 The following statements relating to an object-oriented environment are either true or false. Write the word **true** or **false** beside each statement in the table below.

The output from the following pseudo code is 43.
(method) – void incrementNum(int numParam){
numParam++
}
(main method) – void main(){
int numArg = 42
incrementNum(numArg)
output numArg // writes 43
}	

Object is the base building block of an object-oriented system and all classes are derived from it.	
Overloading occurs where two methods have the same name and take the same list of parameters, even if they return different types.	
A derived / sub class inherits the properties, methods and events of the super / base class and can be customised with additional properties and methods.	
C# / Java implement multiple inheritance through the use of Interfaces.	
A primary concept of object-oriented programming is Polymorphism. It allows sub / derived class methods to be invoked through a super / base class reference during run-time. This is enabled through early binding.	

[6]

Examiner Only	
Marks	Remark

BLANK PAGE

(Questions continue overleaf)

Examiner Only	
Marks	Remark

2 Decisions are one of the basic building blocks of structured software solutions. They allow branching to alternative processes. Write the code which would apply to the following scenarios. Declare and use meaningful variable names.

(a) Set a discount rate to 10% if a sale amount is greater than or equal to £75.00 and the customer does not have a rating of 'C'.

[5]

(b) Check if a sentence contains the word **" the "** and:

If true –

- Replace all occurrences of the word " the " with " a " or replace with " an " if the following word starts with a vowel.
- Output the modified sentence.

If false –

- Output the words "No Change".

Note. Assume the sentence does not begin with the word "the" and that only one space exists between each word.

Example: "A cow jumped over the anvil and the fence."

Examiner Only	
Marks	Remark

[11]

Examiner Only	
Marks	Remark

3 A company, Mourne Mountain Climbers, offers four packages to their clients with charges as shown in the table below.

Bookings are taken detailing the booking number, name of client, package type and the number in the party. A deposit of 10% of the booking cost is taken to confirm the booking and the booking is not recorded until the appropriate deposit is paid.

Package	**Charge per person £**
A : Two starter lessons	50.00
B : Three hour cliff climb with lesson	75.00
C : One day mountain climb trip	125.00
D : One week expert climbing tour	395.00

Note. If the client requires package C or package D and they have a party of 6 or more, a 15% surcharge is made to cover extra staff and fuel for a larger mini-bus.

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

Ensure the creation of:

- A constructor method with fields;
- **GET** and **SET** (Properties / Methods) for the field **noInParty** only;
- **Methods** to determine appropriate costs.

```
class Booking{
    private     int         bookingNo;
    private     String      clientName;
    private     char        package;
    private     DateTime    bookingDate;
    private     int         noInParty;

    public Booking(){
            bookingNo = 0;
            clientName = null;
            package = ' ';
            bookingDate = new DateTime();
            noInParty = 0;
    }
```

// field / parameterised constructor

[3]

// GET and SET (Properties / Methods) for the field **noInParty** only

[4]

Examiner Only	
Marks	Remark

// A method to determine the **cost** due for the booking.

[13]

// A method to determine the **deposit** due for the booking.

[2]

Examiner Only	
Marks	Remark

// A method to determine the **outstanding Cost** due for the booking.

[2]

(b) A program using the class Booking has the following line of code:

Booking booking = **new Booking**(1000, "Joe Bloggs", 'C', **DateTime.Now**,10);

(i) Explain the terms in bold.

[2]

(ii) Write the **code** which will output the outstanding cost due for Joe Bloggs.

[1]

Examiner Only	
Marks	Remark

Examiner Only	
Marks	Remark

4 **(a)** Testing is an important stage of software development. The data used and the processing of this data must be correct.

All data entered should be tested to ensure correctness before storage and use.

(i) Name **three** types of check which should be considered for data entry.

[3]

(ii) For **one** of the Booking class fields defined below from **Question 3**, indicate which checks should be applied to test the validity.

```
class Booking{
    private     int         bookingNo;
    private     String      clientName;
    private     char        package;
    private     DateTime    bookingDate;
    private     int         noInParty;
```

[1]

(b) Exception Handling is a feature which can be implemented to ensure that an object cannot be instantiated or updated if its data is invalid.

Examiner Only	
Marks	Remark

(i) Describe how Exception Handling is implemented in the class design and specify any further classes required.

[6]

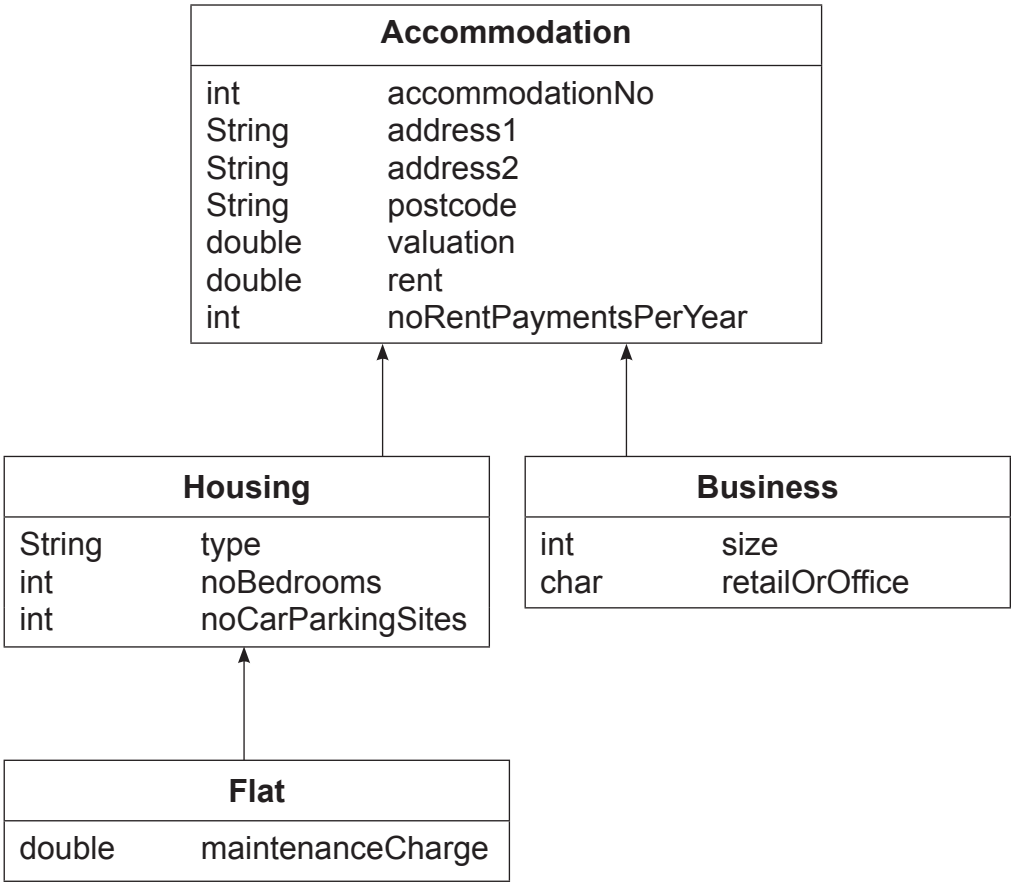
(ii) Describe how this feature would be applied to prevent a program crash when an invalid value is entered for a field in the object.

[3]

Examiner Only	
Marks	Remark

5 A property developer, Peter McKenna, has a portfolio of various types of accommodation for rent. He currently holds details of detached houses, semi-detached houses, terraced houses, flats and business premises on his books.

Inheritance diagram for PMK Rentals.



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

abstract class Accommodation

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

_____

_____

_____

_____ [3]

(b) Assume the class **Accommodation** has been designed with the following elements:

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

Write the code for an additional method, **incomeYear()**, that will return the annual income.

Annual income is calculated as rent multiplied by noRentPaymentsPerYear.

_____

_____

_____

_____ [2]

Examiner Only	
Marks	Remark

(c) (i) Write the code for the derived class **Flat** giving the following:

- class header;
- field definitions;
- field / parameterised constructor;
- GET and SET (Properties / Methods).

[9]

Examiner Only	
Marks	Remark

(ii) Write the method incomeYear() which will calculate and return the total annual cost for a flat.

[4]

(iii) What term should be applied to this method to ensure the correct income is automatically calculated for objects of type Flat.

[1]

(d) Given the following array of properties held for PMK Rentals, write the section of code which will calculate and output the total annual income for the fifty properties.

```
Accommodation [ ] accArray = new Accommodation[50];
```

[5]

Examiner Only	
Marks	Remark

Examiner Only	
Marks	Remark

6 In the hierarchical structure defined for PMK Rentals, **Question 5**, the header for the class **Accommodation** has been amended as follows:

C#
abstract class Accommodation **: IComparable**
Java
abstract class Accommodation **implements Comparable**

(a) Explain the term in bold and indicate why it would be applied to the class.

[4]

(b) The method

C# CompareTo(Object obj)
Java compareTo(Object obj)

which returns an integer, must be implemented in the class Accommodation. Write the code for this method that will compare two rent amounts.

[5]

(c) Mr. McKenna requires a list of his properties in order of the rent charged. Write the code that will **sort** the array accArray and **output** the property details.

Assume the toString( ) method has been implemented for all classes.

Accommodation [ ] accArray = new Accommodation[50];

[5]

Examiner Only	
Marks	Remark

THIS IS THE END OF THE QUESTION PAPER

215547