GCE AS – ⬚ NEW AS

wjec cbac

B500U20-1

eduqas
Part of WJEC

COMPUTER SCIENCE – Component 2
Practical Programming to Solve Problems

P.M. TUESDAY, 14 June 2016

2 hours 15 minutes

INSTRUCTIONS TO CANDIDATES

Answer **ALL** of questions 1, 2 and 3.

Answer only **ONE** section of question 4. This is the section which requires you to use the Integrated Development Environment (IDE) of your chosen programming language.

You will need to record all of your answers to questions 1, 2 and 3 in a word processed document.

INFORMATION FOR CANDIDATES

The number of marks is given in brackets at the end of each question or part-question.

You are reminded of the need for good English and orderly, clear presentation in your answers.
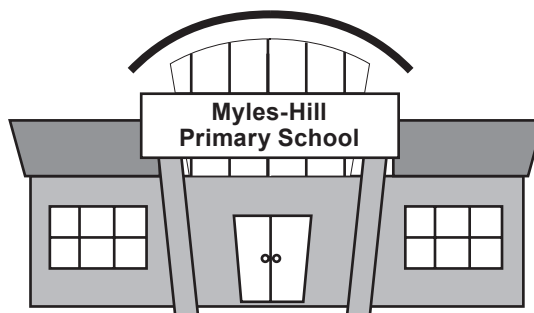
The total number of marks available is 60.

You will need a computer with an installed functional copy of the Integrated Development Environment (IDE) appropriate to your chosen programming language and word processing software.

A calculator is allowed in this examination.

Remember to save your work regularly.

BLANK PAGE

Scenario

The Headteacher of **Myles-Hill Primary School** has decided to commission a new computerised system to store details about the school's pupils and staff.

The school intends to store pupil details such as pupil ID, first name, surname, home address, telephone number and emergency contact.
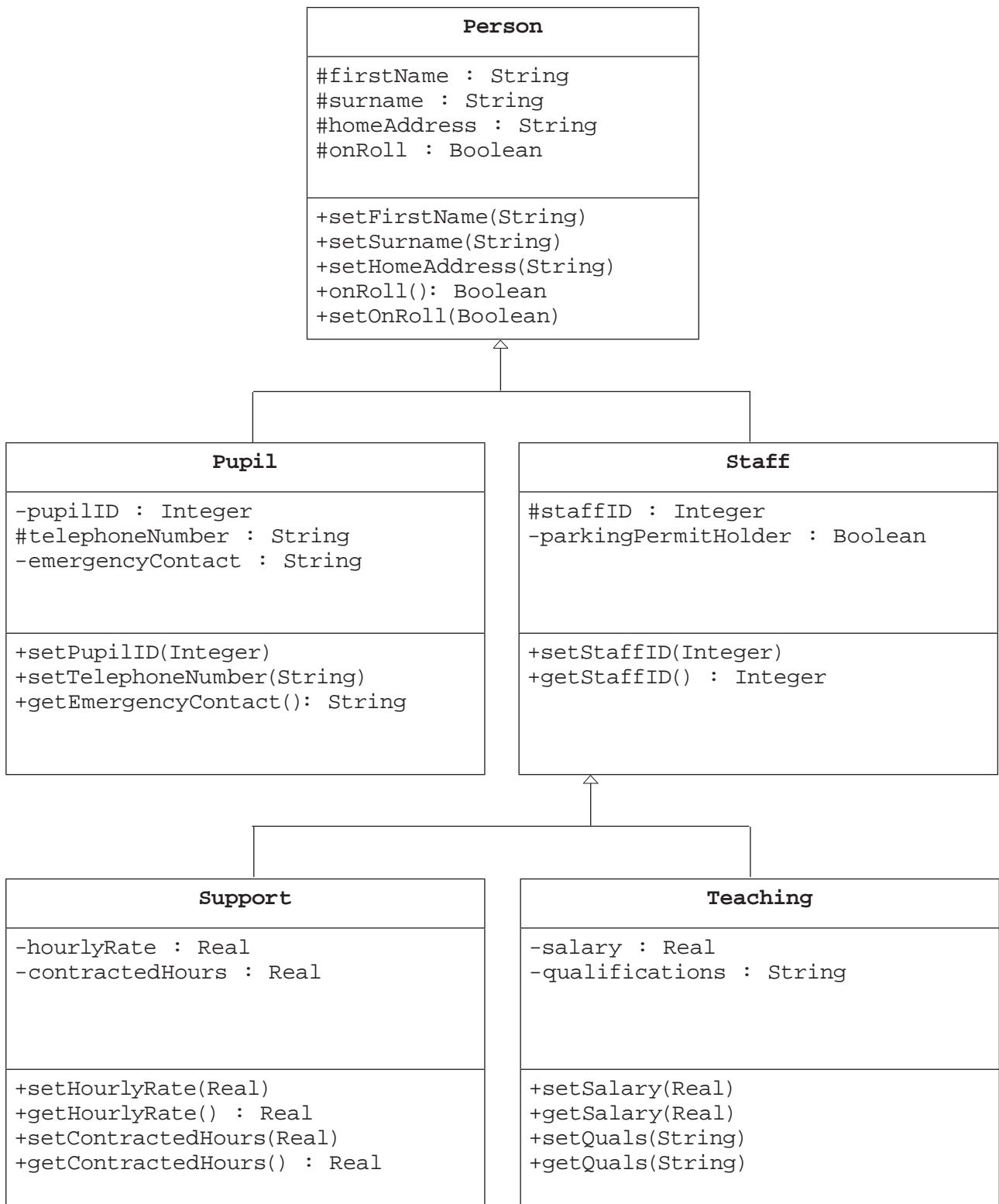
For staff details, the school intends to store staff ID, first name, surname, home address and whether the staff member has purchased a parking permit, which allows the staff member to park on the school premises.

The school will store different details for teaching staff and support staff. For example, the school will need to store the professional qualifications of a teaching staff member and their salary. The school will store the hourly wage and contracted hours of the support staff member.

Records will not be deleted from the system, however, the school will need to log whether the pupil or staff member is currently on roll.

You have been commissioned to develop this new computer system.

1. The following partially complete class diagram represents the relationships between some of the classes that **Myles-Hill Primary School** has decided to create.

```
┌─────────────────────────────────────┐
│               Person                │
├─────────────────────────────────────┤
│ #firstName : String                 │
│ #surname : String                   │
│ #homeAddress : String               │
│ #onRoll : Boolean                   │
├─────────────────────────────────────┤
│ +setFirstName(String)               │
│ +setSurname(String)                 │
│ +setHomeAddress(String)             │
│ +onRoll(): Boolean                  │
│ +setOnRoll(Boolean)                 │
└─────────────────────────────────────┘
```

```
┌──────────────────────────────────┐     ┌──────────────────────────────────┐
│              Pupil               │     │              Staff               │
├──────────────────────────────────┤     ├──────────────────────────────────┤
│ -pupilID : Integer               │     │ #staffID : Integer               │
│ #telephoneNumber : String        │     │ -parkingPermitHolder : Boolean   │
│ -emergencyContact : String       │     │                                  │
├──────────────────────────────────┤     ├──────────────────────────────────┤
│ +setPupilID(Integer)             │     │ +setStaffID(Integer)             │
│ +setTelephoneNumber(String)      │     │ +getStaffID() : Integer          │
│ +getEmergencyContact(): String   │     │                                  │
└──────────────────────────────────┘     └──────────────────────────────────┘
```

```
┌──────────────────────────────────┐     ┌──────────────────────────────────┐
│             Support              │     │             Teaching             │
├──────────────────────────────────┤     ├──────────────────────────────────┤
│ -hourlyRate : Real               │     │ -salary : Real                   │
│ -contractedHours : Real          │     │ -qualifications : String         │
│                                  │     │                                  │
├──────────────────────────────────┤     ├──────────────────────────────────┤
│ +setHourlyRate(Real)             │     │ +setSalary(Real)                 │
│ +getHourlyRate() : Real          │     │ +getSalary(Real)                 │
│ +setContractedHours(Real)        │     │ +setQuals(String)                │
│ +getContractedHours() : Real     │     │ +getQuals(String)                │
└──────────────────────────────────┘     └──────────────────────────────────┘
```

(a) State the name of an identifier for a subclass of `Staff`. [1]

(b) State the return data type for the method `getEmergencyContact`. [1]

(c) State the name of the most appropriate identifier for a variable that is used to store the following:

   (i)   FALSE [1]

   (ii) 37.5 [1]

(d) Explain the significance and implications of the "-" in the property "`-pupilID : Integer`" in the class `Pupil`. [3]

(e) An object called `assistant` is instantiated from the class `Support`.

Applying the concept of inheritance, list **all** of the methods that would be contained within the object `assistant`. [3]

(f) An object called `pupilA` is instantiated from the class `Pupil`.

Applying the concept of inheritance, list **all** of the attributes that would be accessible within the object `pupilA`. [2]

(g) Explain why the property `firstName` is situated within the class `Person`. [2]

(h) Giving an example, describe what would be needed to allow the `parkingPermitHolder` property within the `Staff` class to be set or read by an object from a different class and explain why this is needed. [5]

Save your work for Question 1 as a word processed document named Question 1.

2. **Myles-Hill Primary School** uses an algorithm to calculate a reading score, indicating how suitable a passage of text is for their pupils.

The reading score is calculated as shown below:

Stage 1
- The total number of words in a passage of text is divided by the total number of sentences in that passage.

Stage 2
- The total number of syllables in a passage of text is divided by the total number of words in that passage.

Stage 3
- The reading score is calculated by subtracting the result of the Stage 2 calculation from the Stage 1 calculation.

(a) Using the algorithm above, calculate the reading score for a passage of text where there are 60 syllables, 40 words and 4 sentences. [1]

(b) In another example, where the reading score is calculated to **one decimal place**, the division of two of the input numbers may lead to a result that may have to be truncated.

Using the algorithm above and giving appropriate worked examples, describe truncation and its effects upon the accuracy of the reading score. [5]

(c) Below is an improved algorithm, which calculates a reading score for a passage of text.

```
1  declare readingscore[1..99] is real {array}
2  syllablecount is integer
3  wordcount is integer
4  sentencecount is integer
5  paragraphcount is integer
6  wordsinsentence is real
7  syllablewords is real
8  flag is boolean
9  set flag = false

10 input paragraphcount

11 for i = 1 to paragraphcount

12  input syllablecount
13  input wordcount
14  input sentencecount

15  wordsinsentence = wordcount / sentencecount  {divide}
16  syllablewords = syllablecount / wordcount {divide}

17  readingscore[i] =  wordsinsentence - syllablewords

18  output readingscore[i]

19  if readingscore [i] > 8 then
20   flag = true
21   output "Difficult paragraph"
22  end if

23 next i
```

Copy and **complete** the table below to show how each variable changes when the algorithm is tested by dry-running the test data given below. You may write answers correct to one decimal place. [4]

Test data:
```
Number of paragraphs: 4
1st paragraph: 70 syllables, 30 words, 4 sentences.
2nd paragraph: 80 syllables, 45 words, 5 sentences.
3rd paragraph: 50 syllables, 35 words, 5 sentences.
4th paragraph: 60 syllables, 40 words, 4 sentences.
```

i	wordsinsentence	syllablewords	readingscore[i]	flag
1	7.5	2.3	5.2	false
2	9	1.8	7.2	false
3	7	1.4	5.6	false
4	10	1.5	8.5	true

Save your work for Question 2 as a word processed document named Question 2.

3. **Myles-Hill Primary School** is staging their annual sports day.

Using a recognised convention, design an algorithm to help process the results of a race. There are eight lanes in the race and the finish time for each competitor is recorded in order of lane number.

Your algorithm should contain the following inputs:

• The finish times for each of the eight lanes

Your algorithm should output appropriate labels with the times and corresponding lane numbers of the:

• Gold medal winner (fastest time)
• Silver medal winner (second fastest time)
• Bronze medal winner (third fastest time)

Your algorithm should be written using self-documenting identifiers. [11]

Save your work for Question 3 as a word processed document named Question 3.

4. Select the programming language of your choice from section *(a)*, *(b)* or *(c)* and answer **all** questions in your chosen section.

   (a) **Visual Basic**

   Myles-Hill Primary School wants a computer system to be developed using **Visual Basic** that meets the requirements outlined below:

   - The ability to store pupil details
   - The ability to store staff details
   - The ability to count the number of teachers with particular attributes, e.g. the total number of "English specialist" teaching staff that are available
   - The ability to search staff details

   (i) Open the file Pupils.sln
      - Read through the code and familiarise yourself with its contents
      - The file contains incomplete code, which is intended to save pupil details and return the number of pupils that match the search criteria entered

   Complete this code. [4]

   Remember to save the changes made to the file Pupils.sln

   (ii) Create a new form that will allow **Myles-Hill Primary School** to:

      - Input staff details
      - Validate staff details
      - Store staff details on disc in a text file called staffdetails.txt
      - Confirm storage of staff details in a text file called staffdetails.txt
      - Retrieve specified staff details from disc [12]

   Save your work as staffDetails.frm

   (iii) Using the internal facility of your chosen language, add annotated listings to your code from question 4*(a)*(ii) that would clearly explain the design of your program to another software developer. [4]

   Save your annotations in the same file as 4*(a)*(ii) above.

(b) **Java**

Myles-Hill Primary School wants a computer system to be developed using **Java** that meets the requirements outlined below:

- The ability to store pupil details
- The ability to store staff details
- The ability to count the number of teachers with particular attributes, e.g. the total number of "English specialist" teaching staff that are available
- The ability to search staff details

(i) Open the file Pupil.java
- Read through the code and familiarise yourself with its contents
- The file contains incomplete code, which is intended to save pupil details and return the number of pupils that match the search criteria entered

Complete this code. [4]

Remember to save the changes made to the file Pupil.java

(ii) Create a new class that will allow **Myles-Hill Primary School** to:

- Input staff details
- Validate staff details
- Store staff details on disc in a text file called staffdetails.txt
- Confirm storage of staff details in a text file called staffdetails.txt
- Retrieve specified staff details from disc [12]

Save your work as staffDetails.java

(iii) Using the internal facility of your chosen language, add annotated listings to your code from question 4*(b)*(ii) that would clearly explain the design of your program to another software developer. [4]

Save your annotations in the same file as 4*(b)*(ii) above.

(c) **Python**

Myles-Hill Primary School wants a computer system to be developed using **Python** that meets the requirements outlined below:

- The ability to store pupil details
- The ability to store staff details
- The ability to count the number of teachers with particular attributes, e.g. the total number of "English specialist" teaching staff that are available
- The ability to search staff details

(i) Open the file Pupils.py
- Read through the code and familiarise yourself with its contents
- The file contains incomplete code, which is intended to save pupil details and return the number of pupils that match the search criteria entered

Complete this code. [4]

Remember to save the changes made to the file Pupils.py

(ii) Create a new file that will allow **Myles-Hill Primary School** to:

- Input staff details
- Validate staff details
- Store staff details on disc in a text file called staffdetails.txt
- Confirm storage of staff details in a text file called staffdetails.txt
- Retrieve specified staff details from disc [12]

Save your work as staffDetails.py

(iii) Using the internal facility of your chosen language, add annotated listings to your code from question 4*(c)*(ii) that would clearly explain the design of your program to another software developer. [4]

Save your annotations in the same file as 4*(c)*(ii) above.

END OF PAPER