UNIVERSITY OF
WESTMINSTER

IMPORTANT: The exam paper is provided in two versions according to the Course you are enrolled on.
Answer only the questions in the version corresponding to the programming language that you have been taught during the module:

1) Java Version (pages 2 - 4): BSc Computer Science, BSc Multimedia Computing, BSc Digital Media Development or BEng Software Engineering

2) C++ Version (pages 5 - 7): BSc Computer Games Development

DO NOT TURN OVER THIS PAGE
UNTIL THE INVIGILATOR INSTRUCTS YOU TO DO SO

Section 1

Answer all the following questions in this section if you are enrolled on **BSc Computer Science, BSc Multimedia Computing or BEng Software Engineering**

1) Define a UML class diagram, of the problem presented below:

A cinema has a name and consists of different screens. Each screen has a name and a number of seats. If a cinema stops to exist, and as a result, it needs to be erased, then the screens should also be erased.

A person working in the cinema has a name and a unique ID and can be a member of staff or a projectionist. Each projectionist is responsible for some screens.

(12 points)

2) Briefly describe the access modifiers and how they are represented in the UML diagram.

(9 points)

3) Consider the following class definition:

```
public class CinemaScreen {
      public String name;
      public int seatsNum; // from 1 to 100
};
```

a. Redesigned CinemaScreen class using more appropriate access modifiers according the encapsulation principle.
b. Provide a constructor for the CinemaScreen class that will initialise the instance variables to suitable (valid) start values.
c. Provide signatures for set and get methods for the instance variables.
d. Write bodies for the methods for which you provided signatures in question (c), above. Note that for the set method, the implementation must prevent the instance variable from being set to invalid values.
e. Write a main method that instantiates the radio class and use the set and get methods you designed

(28 points)

4) Suppose that `class Cat` and `class Dog` are subclasses of `class Animal`. Which of the following are legal?

```
a. Dog d = new Animal();
b. Animal a = new Dog ();
c. Dog d = new Cat();
```

(5 points)

5) Consider the following class named Book and reply the following questions.

```
public class Book {

    private String name;
    private String author;
    private int pageNumber;

    public Book(String name){
        this.name = name;
    }

    public void setAuthor(String author){
        this.author = author;
    }

    public void setPage(int pageNumber){
        this.pageNumber = pageNumber;
    }

    public String getAuthor(){
        return author;
    }

    public String getName(){
        return name;
    }


    public int getPage(){
        return pageNumber;
    }


}
```

a. Explain how you could compare different books in Java according to the number of pages and provide an implementation for that.
b. Provide a main class where more than one book are instantiated and stored in an array (or a list) of Books. For each book, set the fields with some values. Print on the screen the ordered list according to the number of pages

(24 points)

6) Explain two methods on how you can terminate a thread in Java. You can provide also examples in java code.

(10 points)

7) Explain briefly what is encapsulation in object oriented programming.

(6 points)

8) What are the differences between error, exception and runtime in java?

(6 points)

Section 2

Answer all the following questions in this section if you are enrolled on **BSc Computer Game Development**

1) Define a UML class diagram, of the problem presented below:

A cinema has a name and consists of different screens. Each screen has a name and a number of seats. If a cinema stops to exist, and as a result, it needs to be erased, then the screens should also be erased.

A person working in the cinema has a name and a unique ID and can be a member of staff or a projectionist. Each projectionist is responsible for some screens.

(12 points)

2) Briefly describe the access modifiers and how they are represented in the UML diagram.

(9 points)

3) Consider the following class definition:

```
public class CinemaScreen {
      public String name;
      public int seatsNum; // from 1 to 100
};
```

4) For all parts of this question, consider a class representing a CinemaScreen. It should have two data fields:

- name of type string;
- seatsNumber of type int, with only values from 0 to 100 valid.

a. Provide a CinemaScreen class that uses appropriate access modifiers according the encapsulation principle.
b. Provide a constructor for the CinemaScreen class that will initialise the instance variables to suitable (valid) start values.
c. Provide signatures for set and get methods for the instance variables.
d. Write bodies for the methods for which you provided signatures in question (b), above. Note that for the set method, the implementation must prevent the instance variable from being set to invalid values.
e. Write a main method that instantiates the CinemaScreen class and use the set and get methods you designed

(28 points)

5) Suppose that `class Dog` and `class Cat` are subclasses of `class Animal` and that we have the following objects:

```
Animal animal;
Dog dog;
Cat cat;
```

Which of the following are legal?

```
a. Dog* d = &animal;

b. Animal* a = &dog;

c. Cat* c = &dog;
```

(5 points)

6) Consider the following class named Book representing a football team in a league.

```
class Book {
    public:
      String name;
      int pageNumber;

    public:
      Book(string name, int pageNumber){
        this.name = name;
        this. pageNumber = pageNumber;
    };

      void setPage(int pageNumber){
        this. pageNumber = pageNumber;
    };

    String getName(){
        return name;
    };
    int getPage(){
        return pageNumber;
    };
};
```

a. Implement the `operator<` for Book class in order to provide a way to compare different Book by their points in the league first, then by name.
b. Write a main method where some Books are instantiated and stored in an array (or list) of Book. For each Team, choose the name and number of pages and order the list.

(24 points)

7) Explain what is a mutex and the basic mutex operation in C++.

(10 points)

8) Explain briefly what is encapsulation in object oriented programming.

(6 points)

9) Explain the use of try catch in C++ and provide an example.

(6 points)