BCS THE CHARTERED INSTITUTE FOR IT

BCS HIGHER EDUCATION QUALIFICATIONS
BCS Level 5 Diploma in IT

SOFTWARE ENGINEERING

Friday 20th March 2020 – Afternoon

Answer **any** FOUR questions out of SIX. All questions carry equal marks.

Time: TWO hours

Answer any Section A questions you attempt in Answer Book A
Answer any Section B questions you attempt in Answer Book B

The marks given in brackets are **indicative** of the weight given to each part of the question.

Calculators are **NOT** allowed in this examination.

Section A
Answer Section A questions in Answer Book A

A1.

a) i) Explain the meaning of the term **'software crisis'.**

ii) Give **THREE** examples of ways in which software engineering practices can help to alleviate problems associated with the 'software crisis'.

(13 marks)

b) Describe the **SIX** stages of a component-based software engineering process model.

(12 marks)

A2.

a) Development projects often require staff from a range of disciplines and backgrounds. Describe **SIX** typical personnel factors that might be used to help in selecting the appropriate team members for such projects.

(12 marks)

b) Explain how team size and team structure can influence effective communications in large development teams.

(8 marks)

c) Give **ONE** advantage and **ONE** disadvantage of using 'benchmarking' to measure project productivity.

(5 marks)

A3.

a) Describe **THREE** of the planning documents that you would expect to find in a software development project plan.

(9 marks)

b) i) Explain the purpose of carrying out a total cost of ownership (TCO) study.

ii) A total cost of ownership (TCO) study is being organised for a system upgrade project that involves a new large-scale enterprise application server. Describe **FOUR** factors that should be considered in the study for this project.

(16 marks)

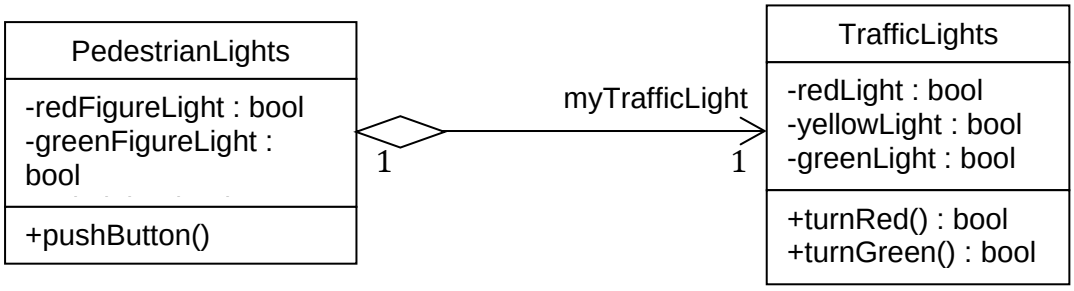
[Turn Over]

Section B
Answer Section B questions in Answer Book B

B4.
Consider a system for operating a pedestrian crossing:

The system will illuminate either a red standing figure or green walking figure to indicate to pedestrians when it is dangerous or safe to cross the road. The system will also illuminate combinations of red, yellow and green lights to signal to road traffic when to stop, and when it is safe to proceed. To operate the crossing, a control box is provided with a button for pedestrians to press. The system illuminates the word "Wait" on the control box from the time the button is pressed until it is safe to cross the road.

The design of the control system comprises two classes as shown on the design class diagram below: One class controls illumination of the pedestrian lights, and the other controls illumination of the traffic lights. When a pedestrian (actor) pushes a button on the physical interface, the pushButton() method is called. The boolean values of the attributes of the two classes correspond to the state of illumination of the physical lights on the crossing.



The system operates according to the following rules:

- When the crossing is inactive, pedestrian lights normally have their red standing figure illuminated; the traffic lights normally have their green light illuminated.
- The cycle of the traffic lights is:
  green → yellow → red → red + yellow → green
- The cycle of the pedestrian lights is:
  red figure → red figure + wait → green figure → red figure
- As the pedestrian and traffic lights are changing, each colour (or combination of colours) is illuminated for a pre-set length of time.
- The pedestrian lights request that the traffic lights change, and then change themselves when they receive a response. This communication is synchronous.
- After changing, the traffic lights must be showing green for at least 3 minutes before they can change back to red again.

a) Draw **TWO** UML state transition diagrams, one for **EACH** of the classes. Your diagrams should be consistent with the class diagram above. **(12 marks)**

b) Draw a UML Sequence Diagram which models the process of a pedestrian using the system to cross a road. Your diagram should be consistent with both the class diagram and the state transition diagrams given in answer to part a). **(8 marks)**

c) Using examples from the above scenario, explain the difference between static and dynamic design models. **(5 marks)**

[Turn Over]

B5.

a) In the context of Component Based Software Engineering (CBSE), define the key characteristics of a software component.

(5 marks)

b) Discuss the potential benefits and risks of CBSE as an approach to software re-use.

(8 marks)

c) A team is planning to develop a new software system by integrating several large-scale Commercial Off The Shelf (COTS) software products.

i) Discuss the potential integration problems the team might encounter with such a system.

ii) Outline the design decisions the team will need to consider with respect to selecting and integrating the COTS products.

(12 marks)

B6.

a) Outline the main phases of a generic testing strategy for a medium-sized object oriented software system. As part of your answer, you should describe examples of the types of tests and testing methods that might take place at each phase.

(15 marks)

b) To help improve software quality, a development team wishes to choose between either introducing regular code inspections, or else deploying a static analysis tool. Traditional testing techniques would continue in either case.

Briefly evaluate these **TWO** approaches, suggesting how **EACH** one could be used in the context of a development project, and explaining the potential advantages and disadvantages of each.

(10 marks)

END OF EXAMINATION