

NATIONAL UNIVERSITY OF SINGAPORE

SCHOOL OF COMPUTING

Semester I: 2001-2002

EXAMINATION FOR

CS2103 – SOFTWARE ENGINEERING

November 2001 Time Allowed: 2 Hours

INSTRUCTIONS TO CANDIDATES

1. This examination paper contains **5** questions and comprises **4** printed pages.
2. Answer **ALL** questions.
3. Begin each question on a new page.
4. This is an open-book examination.

QUESTION 1 (15%)

Read the following specification for designing an Elevator software system. Identify actor(s) and use case(s) and draw a use case diagram for the system:

A building has one elevator which can load and upload people of its 11 floors. People press up or down button on floors to call the elevator. Once inside the elevator people press floor button to indicate their choice of floor to stop at. It also displays where is it at any time and the buttons light up to display pending requests. Elevator could be going up or down or waiting for a request. Elevator stops only on requested floors, opens the door and closes them as well when a floor button or close button is pressed or after every 15 seconds. Elevator ignores the move-down requests on its way up and the move-up requests on its way down. When going up it stops at all requested floors until it reaches the uppermost requested floor. It then changes the direction and stops at all requested floors until the lowermost requested floor is served. It stops (and waits at a floor) when all the requests are served.

QUESTION 2 (15%)

Structure the following groups of classes using generalization and aggregation; draw the corresponding class diagram showing the associations and multiplicity. Note that this is not intended to be a complete model, therefore you may ignore the attributes and methods.

Building, Bedroom, Bungalow, OfficeBuilding, Wall, Foundation

QUESTION 3 (25%)

A computer retail company offers several kinds of membership schemes such as ordinary member, valued member, and premium member.

- (a) Using example programs in Java, illustrate and explain an application situation in which the class `Member` **must** be an abstract superclass of `ValuedMember` and `PremiumMember`. (13%)
- (b) Using example programs in Java, illustrate and explain an application situation in which the class `Member` can be a concrete superclass of `ValuedMember` and `PremiumMember`. (12%)

QUESTION 4 (20%)

In Java, the class `Object` serves as the root of the class hierarchy. That is, `Object` is the superclass of every class even if a class definition does not explicitly extend `Object`. For example, the statement

```
class Student
{
    ...
}
```

means

```
class Student extends Object
{
    ...
}
```

By this reasoning, the declaration

```
class Student extends People
{
    ...
}
```

would mean

```
class Student extends Object, People
{
    ...
}
```

But, inheritance of more than one superclass is not allowed in Java.

- (a) How would you resolve the conflict and accommodate both constraints? Explain how and why your method works.

Note: Your answer does not have to be the same as what Java does. So, you don't have to be constrained by what you know about Java programming. (10%)

- (b) What purposes does the root of the class hierarchy, e.g., `Object` in Java, serve? Use examples to explain the purposes. (10%)

QUESTION 5 (25%)

Study the following binary search program.

```
int binarySearch (int[] a, int key)
{
    int low = 0;
    int high = a.length - 1;
    int middle;

    while (low <= high)
    {
        middle = (low + high) / 2;
        if (key == a[middle])
            return middle;
        else if (key < a[middle])
            high = middle - 1;
        else
            low = middle + 1;
    }
}
```

- (a) Draw a flow graph for the binary search program and identify the independent paths. Clearly indicate the instructions that the nodes in the flow graph refer to. (15%)
- (b) From the flow graph, can you identify any potential problem with the program? How would you fix the problem? (10%)

- END OF PAPER -