Name:

_____

_____

Student Number:

_____

Please fill in your Student Number

Student Number : _____

University of Cape Town ~ Department of Computer Science

Computer Science 1016S ~ 2009

Supplementary Test 2

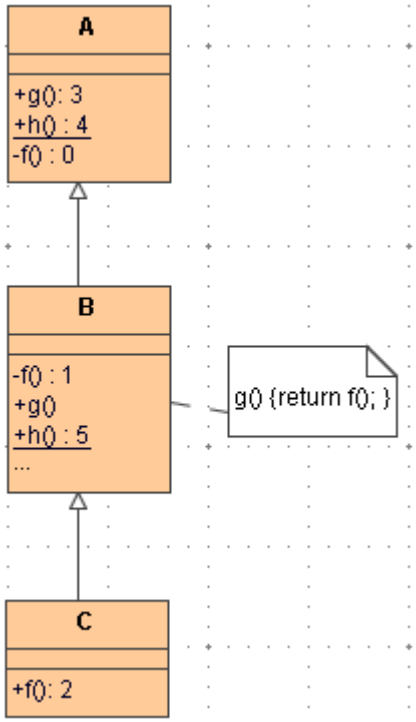
Question	Max	Mark	Internal	External
1	14			
2	10			
3	6			
TOTAL	**30**			

Marks : 30
Time : 40 minutes
Instructions:

a) Answer all questions.
b) Write your answers in the space provided.
c) Show all calculations where applicable.

Question 1: Polymorphism and UML [14 marks]

Use the following UML diagram to answer question 1



a) What kind of relationship between the classes is it? [2]

b) Explain the accessibilities via symbols of a UML class diagram [2]

c) Implement/provide coding of the diagram [6]

d) Which statements below are valid? if a statement is valid, what is the output? [4]

1. C ref1 = new A() _____
2. A ref1 = new C() _____
3. ref1.f() _____
4. ref1.g() _____
5. ref1.h() _____
6. A ref2 = (A) ref1 _____
7. ref2.g() _____
8. ref2.h() _____

Question 2: Cloneable Interface [10 marks]

Given the following code:

```
public class Person implements Cloneable
{
    int personId;
    String personName;
    ..............
        try {
         return super.clone();
        }
        catch (CloneNotSupportedException e) {
         return null;
        }
    ..............
}
```

a) Explain what a **Cloneable** interface is and what it is for? [2]

a) Why is a **CloneNotSupportedException** needed in the above code and when does this exception occur? [2]

b) Given the following code:

```
public class MyClass
{
    int code;
    String name;
    Person  student;
    Person  teacher;
    ............

}
```

Rewrite the above code so that we can make deep copies of **MyClass** objects [6]

Question 3 [6 marks]

Consider the following methods which are in a LinkedList class.

```
/**
   Finds the first node containing the target item, and
   returns a reference to that node. If target is not
   in the list, null is returned.
*/

   private Node<T> find(T target)
   {
       T currentItem;
        while (head != null)
        {
            currentItem = head.data;
            if (currentItem.equals(target))
                return head;
            head = head.link;
        }
        return null; //target was not found
   }

   /**
     Checks whether the list is empty
   */

   public boolean isEmpty( )
   {
      // Fill in
   }
```

a) What is the problem with the find method? Modify it to correct this problem. [4]

b) Fill in the definition for the isEmpty method. [1]

c) What is an iterator? [1]