

Name _____

1. What is a helper method? Given an example. What is a helper method a indication of?
2. What are the three types of messages in Smalltalk. What is the precedence order of the three types of messages?

3. What is the result of executing the following code?

```
Foo>>getBlock  
  | x block |  
  x := 25.  
  block := [:each | each + x].  
  x := 10  
  ^block
```

```
Foo>>runBlock  
  | x block |  
  x := 15.  
  block := self getBlock.  
  x := 20.  
  ^block value: 1
```

4. What is a Template method? Give an example of the use of Template Method in the Smalltalk class library.

5. The scope of a variable is where it can be accessed. That is which methods can access the variable. The lifetime of a variable is how long the variable exists. That is the time from which it is created to the time it no longer exists.

a. What are the scope and lifetime of an instance variable?

b. What are the scope and lifetime of a class instance variable?

6. Describe the Smalltalk naming convention for:

a. Classes

b. Instance variables

c. Arguments to a method

7. Most methods in the Collection class are implemented using three methods `do:`, `add:` and `remove:ifAbsent:`. Show how to implement `select:` in a collection class using the method `do:`. You can return the result in an `OrderedCollection`.

8. We have two classes A and B. What relationship between A and B do the following phrases indicate. Please indicate the role of A and B in the relationship. For example: A is the parent of B.

- a. A is-kind-of B

- a. A has-a B

9. Define the following terms:

a. Abstraction

b. Information Hiding

10. What is Polymorphism? Why is it considered beneficial?

11. Given the classes and methods below what is printed on the Transcript when executing each of the following expressions individually with “do it”?

C new foo

C new bar

B new foo

B new bar

<p>Class A</p> <p>Superclass Object</p> <p>Class Methods</p> <p>new</p> <p> ^super new initialize</p> <p>Instance Methods</p> <p>initialize</p> <p> self print</p> <p>name</p> <p> ^'A'</p> <p>print</p> <p> self print: self name</p> <p>print: aString</p> <p> Transcript show: aString</p>	<p>Class B</p> <p>Superclass A</p> <p>Methods</p> <p>name</p> <p> ^'B'</p> <p>foo</p> <p> self print: self name</p> <p>bar</p> <p> self print: super name</p>
<p>Class C</p> <p>Superclass B</p> <p>Instance Methods</p> <p>initialize</p> <p> super foo</p> <p>name</p> <p> ^'C'</p>	