

# Programming Language Qualifying Exam

## Fall 2005

---

Answer the following 5 problems.

### 1. Variables and Objects

When a program is interpreted or compiled by the computer, the names of the variables, methods, and functions have to be assigned properties, such as memory location, type, and value. The assigning of the property is called *binding*. If it is done at compile-time, it is called *static binding*. If it is done at runtime, it is called *dynamic binding*.

Method names in C++ and Java have both static and dynamic properties.

- (a) Is the type of a method static or dynamic?
- (b) The value (i.e., the actual function that is invoked when a method is called) can be either static or dynamic, and is controlled by the programmer. Using the object oriented language of your choice, give some class definitions that show an example of one method that has a static value, and one method that has a dynamic value. (Note, we are **NOT** talking about the `static` keyword!)
- (c) What is the structure that the compiler / runtime environment uses to implement dynamic values of methods? Explain briefly how it works.

### 2. Abstraction

- (a) A major shift in programming languages and software engineering occurred with the introduction of encapsulated data-types, also known as abstract data-types.
  - i. What are the components of an abstract data-type?
  - ii. What is the cost of an abstract data-type? I.e., what kinds of things are you unable to do with your data if you use this model?
  - iii. What are the benefits of abstract data-types?

### 3. Program Verification

- (a) What is the difference between total correctness and partial correctness?
- (b) Consider the following program.

```
if x < y then m := x;  
if x > y then m := y;
```

The postcondition is supposed to be  $m = \max(x, y)$ , for any inputs  $x$  and  $y$ , but in fact there is an error. Formally demonstrate the error by computing the weakest precondition of this code fragment.

- (c) We want a program that, given an array  $A[0..N]$ , sets the integer  $s$  to be the sum of the entries of the array.
  - i. Write a specification for your program by giving a precondition  $Q$ , postcondition  $P$  and loop invariant  $R$  for the program.

- ii. Write the program, and formally show the conditions in which it satisfies the specification.

#### 4. Parameters

- (a) Consider the following C-like code.

```
int x = 50;

void foo(int i) {
    int a = 10;
    i = i + a;
    i = i + x;
    return;
}

// later we execute...

foo(x);
```

- What will the value of  $x$  be if we use a call-by-value parameter passing style? How is call-by-value implemented?
- What will the value of  $x$  be if we use a call-by-reference parameter passing style? How is call-by-reference implemented?
- What will the value of  $x$  be if we use a call-by-value-result parameter passing style? How is call-by-value-result implemented?

#### 5. Parsing

- (a) Consider the following grammar.

$$\begin{aligned} S &\rightarrow E \$ \\ E &\rightarrow T + E \\ &\quad | \quad T \\ T &\rightarrow F * V \\ &\quad | \quad V \end{aligned}$$

- Give the Characteristic Finite State Machine (CSFM) for the above grammar.
- Is the grammar ambiguous? How can you prove it?
- Is the grammar LL? Why or why not?