

Programming Language Qualifying Exam

Fall 2006

Answer any five of the following six problems. You may attempt all six, and we will use the five highest scoring for your grade.

1. Compilation

- (a) A compiler C translates a program P from language L into machine language. An interpreter I does not translate; rather it directly performs the actions specified by P . The executable produced by C will almost certainly run faster than any interpreted implementation of the same language. Assuming that this is the case, why do we still have interpreters? Give at least two (good) reasons.
- (b) What is the difference between call-by-reference and call-by-value? Explain (briefly! One or two sentences will do!) how they are implemented. Give some code that will behave differently with call-by-reference than with call-by-value, and explain what that difference is.

2. Abstraction

- (a) What are abstract datatypes? Why are they important?
- (b) Explain how the class syntax of either Java or C++ provides for abstract datatypes.

3. Grammars

Consider the following grammar:

$$\begin{array}{lcl} S & \rightarrow & a E a \\ E & \rightarrow & E x E \\ & & | E y E \\ & & | z \end{array}$$

- (a) Construct the Characteristic Finite State Machine for the above grammar.
- (b) Is the above grammar LL? Why or why not?
- (c) Is the above grammar ambiguous? Give a proof with your answer.

4. Weakest Precondition

- (a) Define *weakest precondition*.
- (b) Given program S , and predicates Q and R , suppose that $wp(S, R) = Q$. Suppose also that x represents the current state, and $S(x)$ represents the state after S has run.
 - i. if $x \in Q$, is it necessary that $S(x) \in R$?
 - ii. if $x \notin Q$, is it possible that $S(x) \in R$?
 - iii. What does it mean if $R \equiv true$?
 - iv. What does it mean if $Q \equiv false$?

- (c) Consider the following program S . Let the postcondition $R \equiv m = |1/x|$. Determine formally the conditions under which this program returns the correct answer. Note, there is a problem with this code. Discuss a possible solution to fix this code, giving both an advantage and disadvantage to your approach.

```
if x >= 0 then m := 1/x;  
if x <= 0 then m := -1/x;
```

5. Loop Verification

- (a) In order to verify the correct operation of a loop, you need to check five formulas. What are they?
- (b) Formally prove or disprove the following loop. The postcondition is $s = \sum_{i=0}^{n-1} A[i]$. You will need to determine the loop invariant.

```
s := 0;  
i := 0;  
do i < n -> s, i := s + a[i], i + 1  
od
```

6. Parallelism

- (a) Given two programs P_1 and P_2 which run in parallel, using shared variables to communicate. What does it mean if they are *interference free*?
- (b) What is *interleaving semantics*? How is it related to interference freedom?
- (c) Suppose that P_1 and P_2 run in parallel, using message passing to communicate. What is necessary to guarantee interference freedom in this case?
- (d) What is the difference between deadlock and divergence?