

OHJ-2056, Principles of Programming Languages, Exam, 19.12. 2011.

examiner: Henri Hansen

No calculators, books, or material. Answer on separate paper.

1. Explain, what the following line of C-code is an example of, and give code that has equivalent functionality in some other language:

```
if(x) if(y){cout << "foo";} else cout << "bar";
```

2. Consider the following grammar (<S> is the initial symbol):

```
<S> → <S> + <S> | <S> - <S> | <P>
```

```
<P> → <P> · x | x
```

Show that the grammar is ambiguous.

3. Consider the following code of a C-like language

```
int x = 0;
int f() { return x; }
int g() { int x = 1; return f(); }
```

What do the functions return if the language is C? Let us assume the language is not C, but rather, some dynamically scoped language. What is the answer then?

4. For the previous question, sketch the activation records and call stack for a situation when main calls g(), g() calls f(), just before control returns from f(). Indicate how the situation differs between static and dynamic scoping.

5. The following code is in an imaginary C-like language.

```
int i
int foo(int $ x, int $ y) {
    x = x + 1;
    y = y + i; }
int main(){
    int A[] = {0,1,2};
    i = 1;
    foo(A[i],i);    print(A);}
```

The operator "\$" indicates a parameter passing mechanism. Assuming print(A) prints out the numbers in array A, how can you deduce from this printout, what the parameter passing mechanism in question is?

6. Give a simple example of structures that would not be well-defined under eager evaluation, but which work under lazy evaluation. (For example, in Haskell) Discuss the advantages and disadvantages of such structures.
7. Discuss character encodings and their relation to the type system in a language