

Programming Languages

XX June 20XX

Rules

Time at disposal: 4 h.

Read carefully the following questions. For every questions, provide an answer. Please remember to justify why the results hold by writing the logical flow that lead you to the answers.

Exercises

1. The following program is written using a pseudo-code. Assume that static scoping and call by reference are used. Expressions are evaluated from left to right. What does the following program print? Motivate the answer.

```
{
  int x = 0;

  int A(reference int y) {
    int x=2;
    y=y+1;
    return B(y)+x;
  }

  int B(reference int y){
    int C(reference int y){
      int x=3;
      return A(y)+x+y;
    }
    if (y==1) return C(x)+y;
    else return x+y;
  }
  write (A(x));
}
```

Assuming static scoping is implemented using a display. Show graphically the status of the display and of the activation record stack in the moment when the control enter for the second time into the function A. For every activation record, you can draw just the value of the field used to store the value of the previous display.

2. Describe in your own words what does call by name mean? What does it means that a variable is captured? How do we avoid capturing of variables?

What does the following code print? Assume a language with static scoping and call by name (the `x++` returns the value of the variable `x` and then increments it). The specification of the language says that given an operator \circ , the valuation of the expressions $E_1 \circ E_2$ consist in: i) the valuation of E_1 ; ii) then the valuation of E_2 ; and the application of \circ to the previous two values.

```
{
  int x=5;
  int P(name int m){
    int x=2;
    return m+x;
  }
  write(P(x++) + x);
}
```

Motivate the answer.

3. Describe using your own words the *mark and sweep* technique: what this techniques is used for and what it does. In case the memory is an issue, what is the technique that can be used to perform the mark and sweep without wasting too much memory? What is the difference between mark and sweep compared to mark and compact? What are the pros and cons of these approaches?
4. Describe using your own words the notion of type equality and type compatibility. What does it mean structural type equivalence? What other alternatives are there for type equivalence?

Describe a case of a programming language having two types T_1 and T_2 such that T_1 is compatible with T_2 but not equivalent.