CS4120/4121/5120/5121—Spring 2019
Homework 3
Semantic Analysis
Due: 22 February 2019

0 Updates

- None yet; watch this space.

1 Instructions

1.1 Partners

You may work alone or with *one* partner on this assignment. But remember that the course staff is happy to help with problems you run into. Use Piazza for questions, attend office hours, or set up meetings with any course staff member for help.

1.2 Homework structure

All problems are required of all students.

2 Problems

1. **Symbol tables**

   For each of the following Xi terms, give a typing context in which it type-checks, or explain why no such typing context exists.

   (a) `if (x + 2 == 4) { return x }`

   (b) `while (f(x,y)) x = x + f(y,x)`

   (c) `a: int[] = b z = {3, f}[y]`

2. **Type checking**

   Suppose that function `f` is declared with this signature:

   ```
   f(x: bool, y: bool): int, int[]
   ```

   Show the full typing derivation for the following Xi statement:

   ```
   x:int, _ = f({true,false,true}[1], 0==1-1)
   ```

   To fit the full derivation onto a page, you may split it into subderivations.

3. **Inference rules**

   Suppose Xi were extended with a new `do-until` statement:

`do` *s* `until (`*e*`)`

The `do-until` statement executes the statement *s* until afterwards *e* evaluates to `false`. In particular, *s* is evaluated at least once, and is always evaluated by the time *e* is evaluated. Consequently, the following program is safe, and it prints the string "132".

```
1  use conv
2  use io
3
4  main(args: int[][]) {
5    x := 0;
6    do {
7      y := x + 1;
8      x := y + 1;
9    } until (x * y > 100)
10   println(unparseInt(x * y));
11 }
```

Give a suitable inference rule in the style of the Xi type system specification to describe the typing of this new statement form.

3 Submission

Submit your solution as a PDF file on CMS. This file should contain your name, your NetID, all known issues you have with your solution, and the names of anyone with whom you have discussed the homework.