

Assignment 6
CSCI 3136: Principles of Programming Languages
Due Mar 26, 2018

Banner ID: _____

Name: _____

Banner ID: _____

Name: _____

Banner ID: _____

Name: _____

Assignments are due on the due date before class and have to include this cover page. Plagiarism in assignment answers will not be tolerated. By submitting their answers to this assignment, the authors named above declare that its content is their original work and that they did not use any sources for its preparation other than the class notes, the textbook, and ones explicitly acknowledged in the answers. Any suspected act of plagiarism will be reported to the Faculty's Academic Integrity Officer and possibly to the Senate Discipline Committee. The penalty for academic dishonesty may range from failing the course to expulsion from the university, in accordance with Dalhousie University's regulations regarding academic integrity.

Question 1 (10 marks) Consider the following Haskell program:

```
x, y :: Int
x = 10
y = 20

f :: Int -> Int
f z = z + y

g :: (Int -> Int) -> Int
g f = let y = 5
      in f x

h :: Int
h = let x = 5
     in g f
```

Haskell uses static scoping and deep binding, but let's assume we can specify different binding models for our Haskell compiler to use. What is the value of `h` for each of the four possible combinations of static and dynamic scoping and shallow and deep binding?

Question 2 (5 marks) In class, we discussed how to implement static scoping using a static chain. We discussed how to use the static chain to determine the location of a variable we want to access at runtime. What we did not discuss is how to maintain this static chain when we make recursive calls. Your task in this question is to provide the details of computing the static link for a function that is being called. Specifically, if the called function is `f`, you have to discuss how the program can identify, at runtime, the stack frame associated with the most recent invocation of the immediately enclosing subroutine of `f`.