CS314 Principles of Programming Languages
Written Assignment 1

March 29, 2021

Name: ____

NetID: ____

Instructions:

- The written assignment has a total of **8** points.
- Answer essay questions in 2 to 3 sentences. Longer answers are not needed.
- For partial credit, show all of your work and clearly indicate your answers.
- You can either annotate your solution on this document or put your solution in another text document (e.g. MS Word) with clear marks to label the answer to each question.
- If you do not know how to type λ, just write lambda in English.
- Submit a PDF version of your solution to Sakai (e.g. using the printing function of Word).

OCaml

1. Write the types of the following OCaml expressions or write "type error" if the expression has no type:

   (a) (½ point) `[("I", 4.0); ("R", 0.0); ("S", 1)]`

   (b) (½ point) `fun a -> fun b -> (a b) + 1`

   (c) (½ point)

   ```
   type 'a option = Some of 'a | None
   let f a =
       if a < 0.0 then None else Some a
   ```

2. Provide expressions (without type annotations) that have the following types:

   (a) (½ point) `int -> int list -> bool list`

   (b) (½ point) `'a -> ('a->'b) -> 'b`

   (c) (½ point) Define a function f that when used in the following expression will not produce any type errors:

   ```
   fold_left f ([],0) [5;4;3;2;1]
   ```

   The implementation and type of fold_left are given for reference, below.

```
    let rec fold_left f a l = match l with
        | [] -> a
        | h::t -> fold_left f (f a h) t
```

Lambda Calculus

3. Choose whether the following statements are true or false:

   (a) (½ point) $\lambda x.\lambda y.y\ x$ is α-equivalent to $\lambda f.\lambda n.n\ f$ A. True / B. False

   (b) (½ point) $\lambda y.y\ x$ is α-equivalent to $\lambda x.x\ y$ A. True / B. False

4. (1 point) Reduce the following λ expression to normal form. Show each reduction step. If already in normal form, write "normal form".

$$(\lambda x.x\ (\lambda x.y\ x))\ (\lambda z.z)$$

5. (1 point) Reduce the following λ expression to normal form. Show each reduction step. If already in normal form, write "normal form". Hint: function application is left-associative $x\ y\ z\ =\ (x\ y)\ z$.

$$(\lambda x.\lambda y.x\ \ y\ \ z)\ (\lambda c.c)\ ((\lambda a.a)\ b)$$

6. (1 point) Reduce the following λ expression to normal form. Show each reduction step. If already in normal form, write "normal form". Hint: you may need to perform α-conversion during the evaluation.

$$(\lambda x.(\lambda y.(x\ y)))\ y$$

7. (1 point) Which of the following lambda term has the same semantics as this bit of OCaml code (choose exactly one):

```
let func x = (fun y -> y x) a b
```

   A. $(\lambda y.y\ x)\ a\ b$
   B. $(\lambda x.(\lambda y.y\ x)\ a\ b)$
   C. $(\lambda x.(\lambda y.y\ x))\ a\ b$
   D. $(x(\lambda y.y\ x))\ a\ b$