CS314 Fall 2018

Assignment 7

Submission: **pdf** file through sakai.rutgers.edu

Problem 1 – Scheme Programming

1. As we discussed in class, **let** and **let*** do not add anything to the expressiveness of the language, i.e., they are only a convenient shorthand. For instance,
   `(let ((x v1) (y v2)) e)` can be rewritten as
   `((lambda (x y) e) v1 v2)`.

   How can you rewrite `(let* ((x v1) (y v2) (z v3)) e)` in terms of λ-abstractions and function applications?

2. Use the `map` and `reduce` functions we learned in class to implement function `maxAbsoluteVal` that determines the maximal absolute value of a list of integer numbers. Example

```
(define maxAbsoluteVal
  (lambda (l)
    ... ))
...
(maxAbsoluteVal '(-5 -3 -7 -10 12 8 7)) --> 12
```

Problem 2 – Lambda Calculus

Use α/β-reductions to compute the final answer for the following λ-terms. Your computation ends with a final result if no more reductions can be applied. Does the order in which you apply the β-reduction make a difference whether you can compute a final result? Justify your answer.

1. $(((\lambda x.x)\ (\lambda x.28))\ (\lambda z.z))$

2. $((\lambda x.((\lambda z.((\lambda x.(z\ x))\ 2))\ (\lambda y.(*\ x\ y))))\ 6)$

3. $((\lambda z.\ ((\lambda y.z)\ ((\lambda x.(x\ x))(\lambda x.(x\ x)))))\ 11)$

Problem 3 – Programming in Lambda Calculus

In lecture 16 and 17, we discussed the encoding of logical constants true and false in lambda calculus, together with the implementation of logical operators.

1. Compute the value of ((and true) true) using β-reductions.
2. Define the or operator in lambda calculus. Prove that your definition is correct, i.e., your lambda term for or implements the logical or operation.
3. Define the exor (exclusive or) operator in lambda calculus. Prove that your definition is correct, i.e., your lambda term for exor "implements" the logical exor operation.

Problem 4 – Lambda Calculus and Combinators S & K

Let's assume the S and K combinators:

- $\mathrm{K} \equiv \lambda xy.x$
- $\mathrm{S} \equiv \lambda xyz.((xz)(yz))$

Prove that the identify function $\mathrm{I} \equiv \lambda x.x$ is equivalent to ((S K) K), i.e.,

$$\mathrm{I} \equiv \mathrm{SKK}$$