CS 496: Quiz 4

13 March, 2019

> For this exercise you must assume that when `debug(e)` is evaluated, the environment is printed (as usual) and evaluation **halts**.

1. Consider the following expression in REC

```
let z = 0
in let prod = proc (x) { proc (y) { x*y }}
in letrec f(n) = if zero?(n) then 1 else ((prod n) (f (n-1)))
in (f 10)
```

A `debug` instruction was placed somewhere in the code and it produced the environments below. Where was it placed? Reproduce the entire expression including the missing `debug` expression. Do this for each of the three items below. Note that there may be more than one solution for each item, it suffices to supply just one.

(a)
```
[
    z:=NumVal 0
    prod:=ProcVal(x,Proc(y,Mul(Var x,Var y)),[z:=NumVal 0])
    f:=Rec(n,IfThenElse(Zero?(Var n),
                        Int 1,
                        App(App(Var prod,Var n),App(Var f,Sub(Var n,Int 1)))))
    n:=NumVal 0
]
```

Draw a box around the argument of `debug`:

```
let z = 0
in let prod = proc (x) { proc (y) { x*y }}
in letrec f(n) = if zero?(n) then 1 else ((prod n) (f (n-1)))
in (f 10)
```

(b)
```
[
    z:=NumVal 0
]
```

Draw a box around the argument of `debug`:

```
let z = 0
in let prod = proc (x) { proc (y) { x*y }}
in letrec f(n) = if zero?(n) then 1 else ((prod n) (f (n-1)))
in (f 10)
```

(c)
```
[
    z:=NumVal 0
    x:=NumVal 10
]
```

Draw a box around the argument of `debug`:

```
let z = 0
in let prod = proc (x) { proc (y) { x*y }}
in letrec f(n) = if zero?(n) then 1 else ((prod n) (f (n-1)))
in (f 10)
```

2. What expressed value is obtained from evaluating the following expression?

```
let z = 0
in let prod = proc (x) { proc (y) { x*y }}
in letrec f(n) = if zero?(n) then 1 else ((prod n) (f (n-1)))
in f
```

Write your solution below:

Ok (ProcVal("n", ITE (IsZero(Var "n"),
Int 1,
App(App(Var "prod", Var "n"), App(Var "f", Sub(Var "n", Int 1)))),
ExtendEnvRec("f","n", ITE (IsZero(Var "n"),
Int 1,
App(App(Var "prod", Var "n"), App(Var "f", Sub(Var "n", Int 1)))),
ExtendEnv("prod", ProcVal("x", Proc "y", Mul(Var "x", Var "y")),
ExtendEnv("z", NumVal 0, EmptyEnv)),
ExtendEnv("z", NumVal 0, EmptyEnv)))))

Submission instructions

You must submit a pdf with your solutions. These may be handwritten and then scanned or annotated. One submission per group.