Problem Set 8
COMP301 Spring 2023
Week 8: 24.04.2023 - 28.04.2023

Instructions:

- Submit your answers to the Blackboard PS8 assignment until April 29th Saturday, at 23.59.
- Please use the code boilerplate, which includes several tests for you to see if your code is correct.
- Submit your code and PDF file to BlackBoard as a single zip file *yourIDno_ username.zip*. (Example: *123456_ hcapuk20_ ps8.zip*)

Problem 1:

a). Draw the contour diagram of the following program. Specify declarations and references.

```
let a = 12 in
    proc (a, b)
        let c = proc (b, a) -(b, 3) in
            let d = -(c, a)
                in (c d)
```

b). Consider the expression

```
let z=4 in
    letrec f(x) = if zero?(x) then 1
                              else (f -(x, 1))
                  in (f z)
```

Draw the environment that is passed to `value-of` when the expression `1` is evaluated. Show the intermediate steps in each value-of call and apply-env call. You can skip the value-of calls which `if-exp` is evaluated.

Problem 2:
Part A. Translate this expression into a nameless expression and include your answer in the pdf file.

```
let a = 6 in
  let b = 4 in
     let c = -(a,b) in
        let a = -(b, c) in
           let func = (proc(a) proc(b) proc(c) -(a,-(b,c))) in
(((func a) b) c)
```

Part B. Translate this nameless expression into a PROC expression and include your answer in the pdf file.

```
%let %nameless-var 13 in
 %let %nameless-var 7 in
    %let %nameless-var 12 in
       %let %nameless-var %lexproc -(#2,#0) in
          %let %nameless-var 5 in
             %lexproc -(#0,-(#4, #3))
```

Problem 3:

a). [1]: Extend the `letrec` language to allow the declaration of any number of mututally recursive unary procedures, for example:

```
letrec-m
  even(x) = if zero?(x) then 1 else (odd -(x,1))
  odd(x) = if zero?(x) then 0 else (even -(x,1))
in (odd 13)
```

evaluates to 1 because 13 is an odd number. The `letrec` implementation is given. Make the necessary modifications stated below.

Note 1: Methods that need to be modified are highlighted inside the `LETREC` language source code with some hints. See the following files: `data-structures.rkt`, `environment.rkt`, `interp.rkt` and `lang.rkt`

Note 2: You need to update the following files: `environment.rkt`, `data-structures`, `interp.rkt` and `lang.rkt`.

Note 3: Do not forget to uncomment necessary tests in `test.rkt`.

b). Could `letrec` and `letrec-m` be merged? What would be the necessary changes? Discuss the necessary changes in your report, they can be applied on given zip file as a challenge but not required.

[1] EOPL p.84-85 Exercise 3.32