

Comp 301 Project 2
Burak Yıldırım 72849 – Alkan Akısu 71455

Part – A

- (1) Syntax, Values, Environments, Specification the Behavior of Expressions, and Implementation of Behavior of Expressions.
- (2) Syntax are located in data-structures.rkt
Values are located in test.rkt
Environments are located in environments.rkt
Specification of the Behavior of Expressions are located in lang.rkt
Implementation of Behavior of Expressions are located in interp.rkt

Part – B

2-

$x = 4$

$[x=4] \rho$

$y = 3$

$[y=3]$

$[x=4] \rho$

$z = 6$

$[z=6]$

$[y=3]$

$[x=4] \rho$

Part – C

Expressed Values: Int + Bool + String

Denoted Values: Int + Bool + String

Part – E

Let the top-to-bottom approach be my-cond1 and the bottom-to-top approach is my-cond2. We can understand the correct one by trying different test cases. Mycond1 goes through all conditional checks whereas mycond2 doesn't check the rest of the conditions when it finds a true condition starting from the bottom. When we try a test case where the first condition is false, the second condition is true, the third one gives an erroneous condition inside, and the fourth one is also a true condition. We can demonstrate this situation as below:

```
(let ((x 6) (y 7) (z 8)) (cond  
  
  ((= x 5) 20)      ;false  
  
  ((> x 4) 12)      ;true  
  
  ((x y) 13)        ;error  
  
  ((< y z) 15)      ;true  
  
  (else 11)))
```

In this case, mycond1 will give an error due to the third condition but mycond2 will give 15 as output. This way, we differentiated these two implementations and found the correct one which is mycond1.

Workload Breakdown:

Alkan was responsible for comp-exp, my-cond, custom expression (switch-exp), creation of the test cases, PartE, and the bonus part.

Burak was responsible for PartA, PartB, PartC, bool-exp, str-exp, op-exp, and bonus part. Each team member has successfully completed their responsibilities on time.

References

For the bonus task the link below is used.

<https://docs.racket-lang.org/reference/strings.html#%28def.%28%28quote.%20~23~25kernel%29.%20substring%29%29>