AI Solution: First Lisp Interactions and Problem Solving

Nahyeon Kim

This assignment consists of the reproductions of two sample Lisp sessions that were presented in class, and Lisp code to solve two simple numeric problems, one pertaining to a die and one pertaining to a goat.

Task 1: Mimic the "Basic Forms" Demo

```
[1]> 496
496
[2]> "Common Lisp with Objects"
"Common Lisp with Objects"
[3]> pie

*** - SYSTEM::READ-EVAL-PRINT: variable PIE has no value
The following restarts are available:
USE-VALUE           :R1          Input a value to be used instead of PIE.
STORE-VALUE         :R2          Input a new value for PIE.
ABORT               :R3          Abort main loop
Break 1 [4]> :a
[5]> pi
3.14159265358979323L0
[6]> (+ pi 496)
499.14159265358979323L0
[7]> (+ 2 3 5 7)
17
[8]> (* (+ 3 6 9) (- 8 5))
54
[9]> (double 5)

*** - EVAL: undefined function DOUBLE
The following restarts are available:
USE-VALUE           :R1          Input a value to be used instead of (FDEFINITION 'DOUBLE).
RETRY               :R2          Retry
STORE-VALUE         :R3          Input a new value for (FDEFINITION 'DOUBLE).
ABORT               :R4          Abort main loop
Break 1 [10]> :a
[11]> (quote pie)
PIE
[12]> (quote (double 5))
(DOUBLE 5)
[13]> 'pie
```

```
PIE
[14]> '(double 5)
(DOUBLE 5)
[15]> (setf pie 'cherry)
CHERRY
[16]> pie
CHERRY
[17]> (setf dozen 12)
12
[18]> dozen
12
[19]> (defun double (x) (* x 2))
DOUBLE
[20]> (double dozen)
24
[21]> (double 5)
10
[22]> (double pi)
6.28318530717958647 7L0
[23]> (double pie)

*** - *: CHERRY is not a number
The following restarts are available:
USE-VALUE          :R1          Input a value to be used instead.
ABORT              :R2          Abort main loop
Break 1 [24]> :a
[25]> (bye)
Bye.
```

Task 2: Mimic the "Numeric Forms" Demo

```
[1]> (+ 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20)
210
[2]> (/ (* 20 21) 2)
210
[3]> (+)
0
[4]> (*)
1
[5]> (+ 1 2 3)
6
[6]> (* 1 2 3)
6
[7]> (- 1 2 3)
-4
[8]> (/ 1 2 3)
```

```
1/6
[9]> (mod 15 4)
3
[10]> (mod 4 15)
4
[11]> (sqrt 100)
10
[12]> (sqrt 2)
1.4142135
[13]> (expt 7 60)
508021860739623365322188197652216501772434524836001
[14]> ; circumference of a radius 10 circle
(* 2 pi 10)
62.83185307179586477L0
[15]> ;area of a radius 15 circle
(* pi (expt 17.2 2))
929.4089
[16]> ; area of a ring bounded by contentric circles of radii 15 and 17.2
(- (* pi (expt 17.2 2)) (* pi (expt 15 2)))
222.55052
[17]> (bye)
Bye.
```

Task 3: "Percent of Die that is White" problem

Suppose a standard die measures 3.25cm on the edge of a face. Further, suppose that each dot on the die has a diameter of length one-fifth the edge of a face. What percent of the surface area of the die is white?

Code: "Percent of Die that is White"

```
;-------------------------------------------------------------------------------
;File: White_Percent_of_a_Standard_Die.l
;line: Lisp forms to solve the "White percent of a standard die" problem

;-------------------------------------------------------------------------------
; Problem:
;Suppose a standard die measures 3.25cm on the edge of a face. Further, suppose that each
dot on the die has a diameter of length one-fifth the edge of a face. What percent of the
surface area of the die is white?

;-------------------------------------------------------------------------------
;Solution

;observations
(setf one_face_size (* 3.25 3.25))
```

```
(setf one_dot_size (* (/ 1 5) (/ 3.25 2) pi))

; related deductions
(setf all_face_size (* 6 one_face_size))
(setf all_dot_size (* (+ 1 2 3 4 5 6) one_dot_size))

;compute final result

(setf get_percent (* 100 (/ (- all_face_size all_dot_size) all_face_size)))
```

Demo: “Percent of Die that is White”

```
[1]> (load "White_Percent_of_a_Standard_Die.l")
;; Loading file White_Percent_of_a_Standard_Die.l ...
;; Loaded file White_Percent_of_a_Standard_Die.l
#P"/home/nahyeon/Desktop/Lisp/White_Percent_of_a_Standard_Die.l"
[2]> one_face_size
10.5625
[3]> one_dot_size
1.0210177
[4]> all_face_size
63.375
[5]> all_dot_size
21.441372
[6]> get_percent
66.167465
[7]> (bye)
Bye.
```

Task 4: “Tethered Goat” problem

A goat is tethered to one corner of a barn that is isolated in a huge grassy field. In whatever units, the barn measures 62 x 44, and the rope is of length 88. What is the area of land on which the goat gets to graze?

Code: “Tethered Goat”

```
;-------------------------
; File: Tethered_Goat.l
; Line: Lisp forms to solve the "Tethered Goat" problems
;-------------------
; Problem:
; A goat is tethered to one corner of a barn that is isolated in a huge grassy field. In whatever
units, the barn measures 62 x 44 and the rope is of length 88. What is the area of land on
which the goat gets to graze?

;----------------------------
; Solution
; observations
```

```
(setf whole_field_size (* 88 88 pi))
(setf barn_size (* 44 62))

;compute final result
(setf graze_size (- whole_field_size barn_size))
```

Demo: “Tethered Goat”

```
[1]> (load "Tethered_Goat.l")
;; Loading file Tethered_Goat.l ...
;; Loaded file Tethered_Goat.l
#P"/home/nahyeon/Desktop/Lisp/A2/Tethered_Goat.l"
[2]> whole_field_size
24328.4935093993588 39L0
[3]> barn_size
2728
[4]> graze_size
21600.4935093993588 39L0
[5]> (bye)
Bye.
```