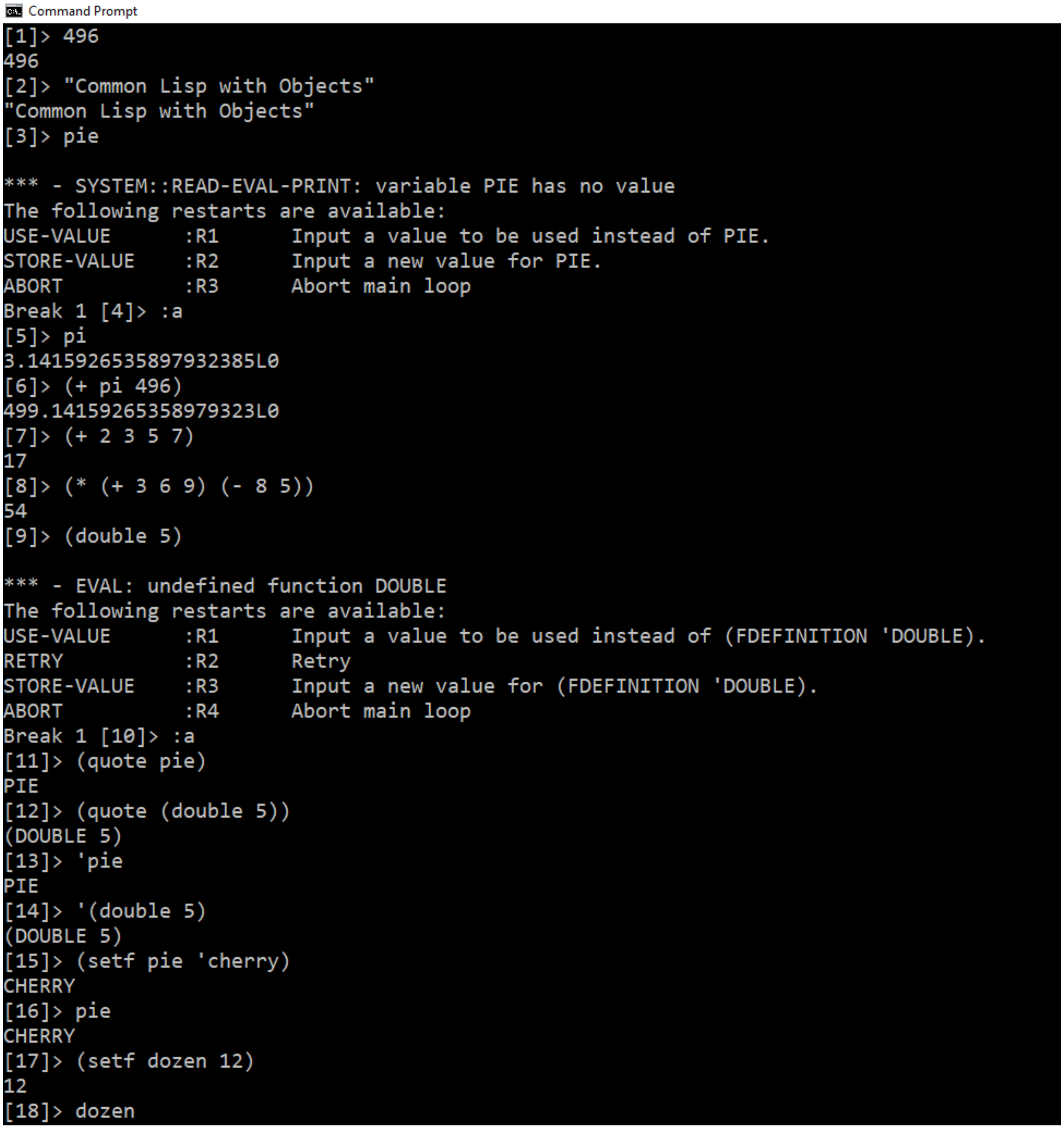
Kuncheng Feng

AI Solution: First Lisp Interactions and Problem Solving

This assignment consists of the reproductions of two sample Lisp sessions presented in class, as well as two additional questions.

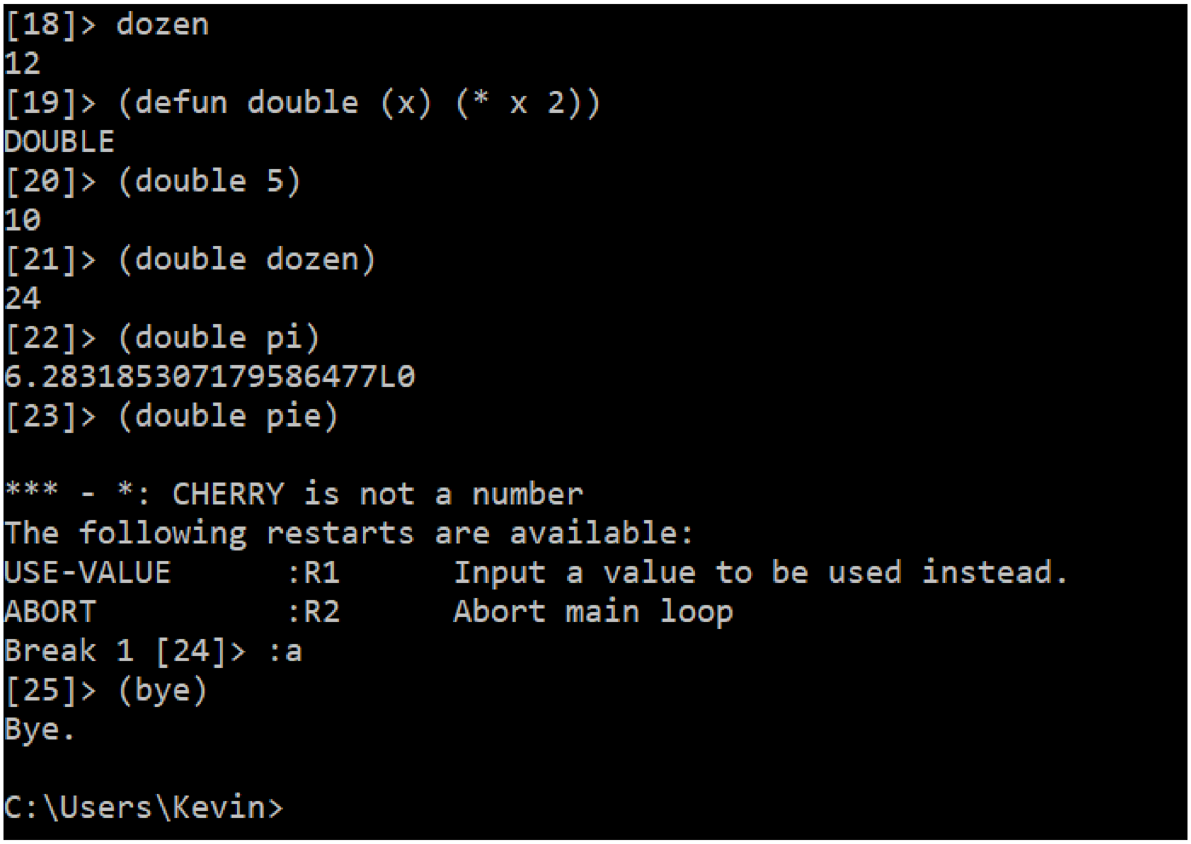
Task 1: Mimic the “Basic Forms” Demo



```
[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
3.1415926535897932385L0
[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
```

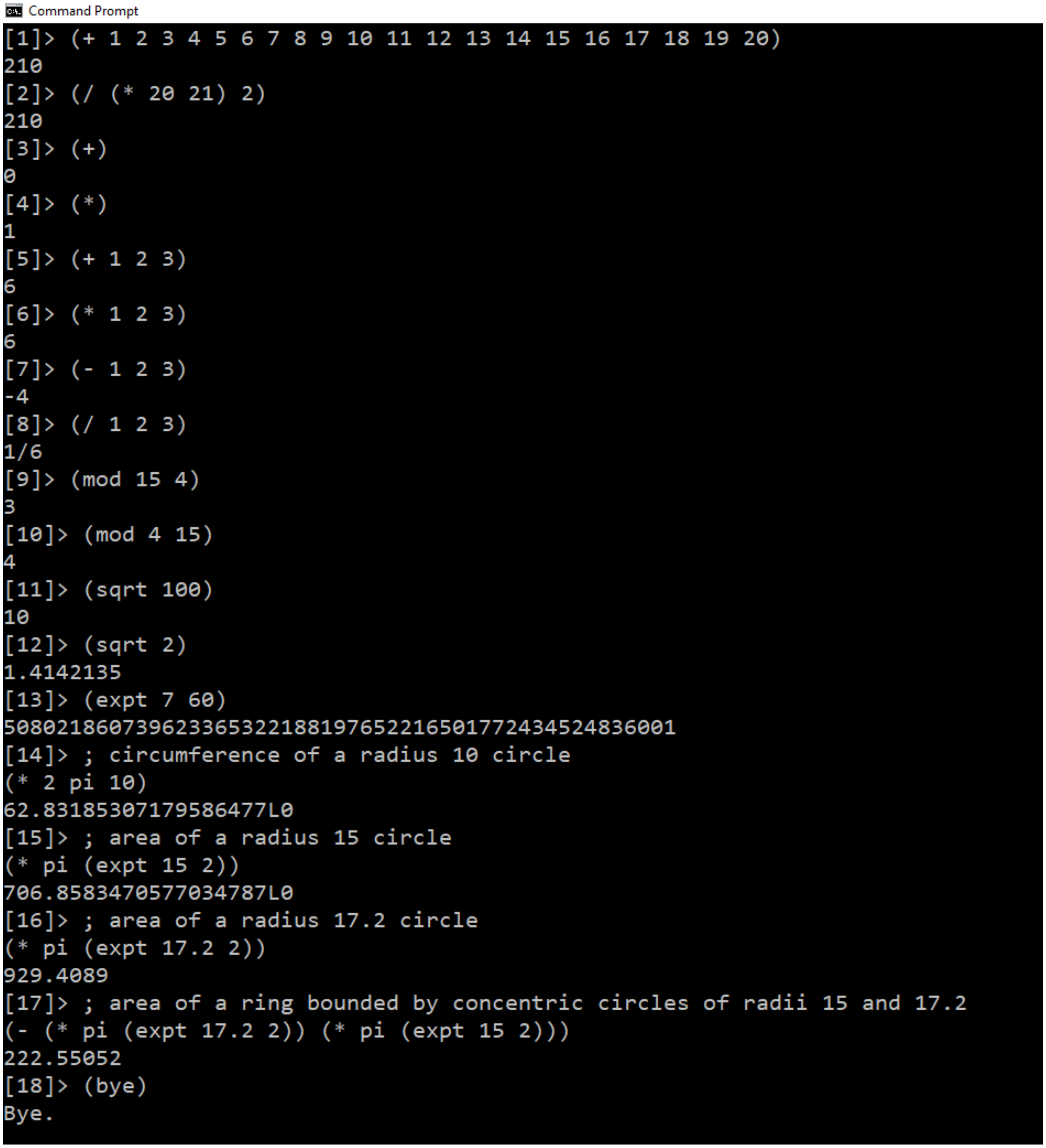


```
[18]> dozen
12
[19]> (defun double (x) (* x 2))
DOUBLE
[20]> (double 5)
10
[21]> (double dozen)
24
[22]> (double pi)
6.283185307179586477L0
[23]> (double pie)

*** - *: 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.

C:\Users\Kevin>
```

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)
62.831853071795864770L0
[15]> ; area of a radius 15 circle
(* pi (expt 15 2))
706.85834705770347870L0
[16]> ; area of a radius 17.2 circle
(* pi (expt 17.2 2))
929.4089
[17]> ; area of a ring bounded by concentric circles of radii 15 and 17.2
(- (* pi (expt 17.2 2)) (* pi (expt 15 2)))
222.55052
[18]> (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 of “Percent of Die that is White”:

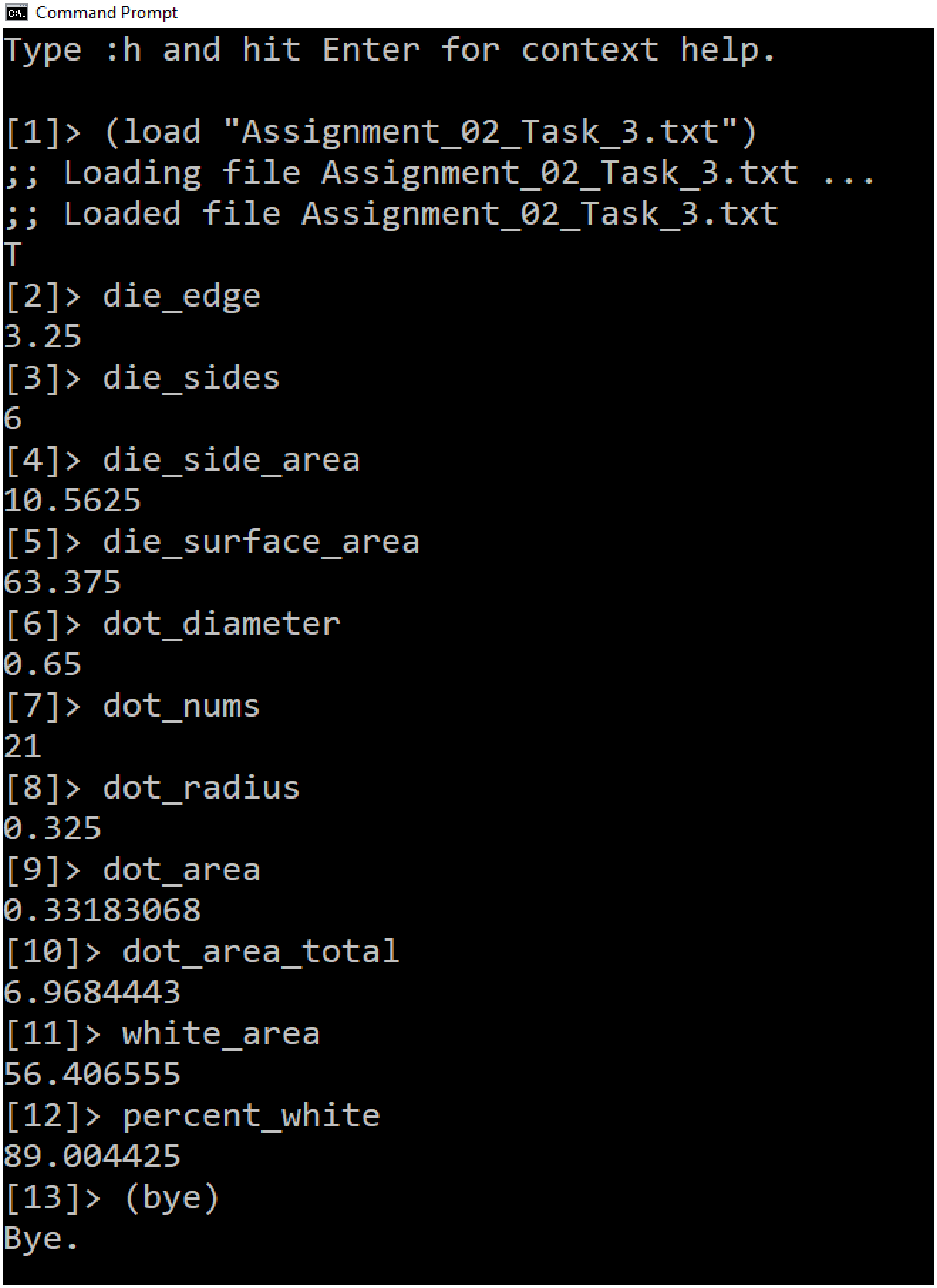
```
; Facts:
     (setf die_edge 3.25)
     (setf die_sides 6)
     (setf dot_diameter (/ die_edge 5))
     (setf dot_nums (+ 1 2 3 4 5 6))

; Deduction on Die:
     (setf die_side_area (* die_edge die_edge))
     (setf die_surface_area (* die_side_area die_sides))

; Deduction on Dot:
     (setf dot_radius (/ dot_diameter 2))
     (setf dot_area (* pi (* dot_radius dot_radius)))
     (setf dot_area_total (* dot_area dot_nums))

; Final Calculation:
     (setf white_area (- die_surface_area dot_area_total))
     (setf percent_white (* (/ white_area die_surface_area)
100))
```

Demo of “Percent of Die that is White”:



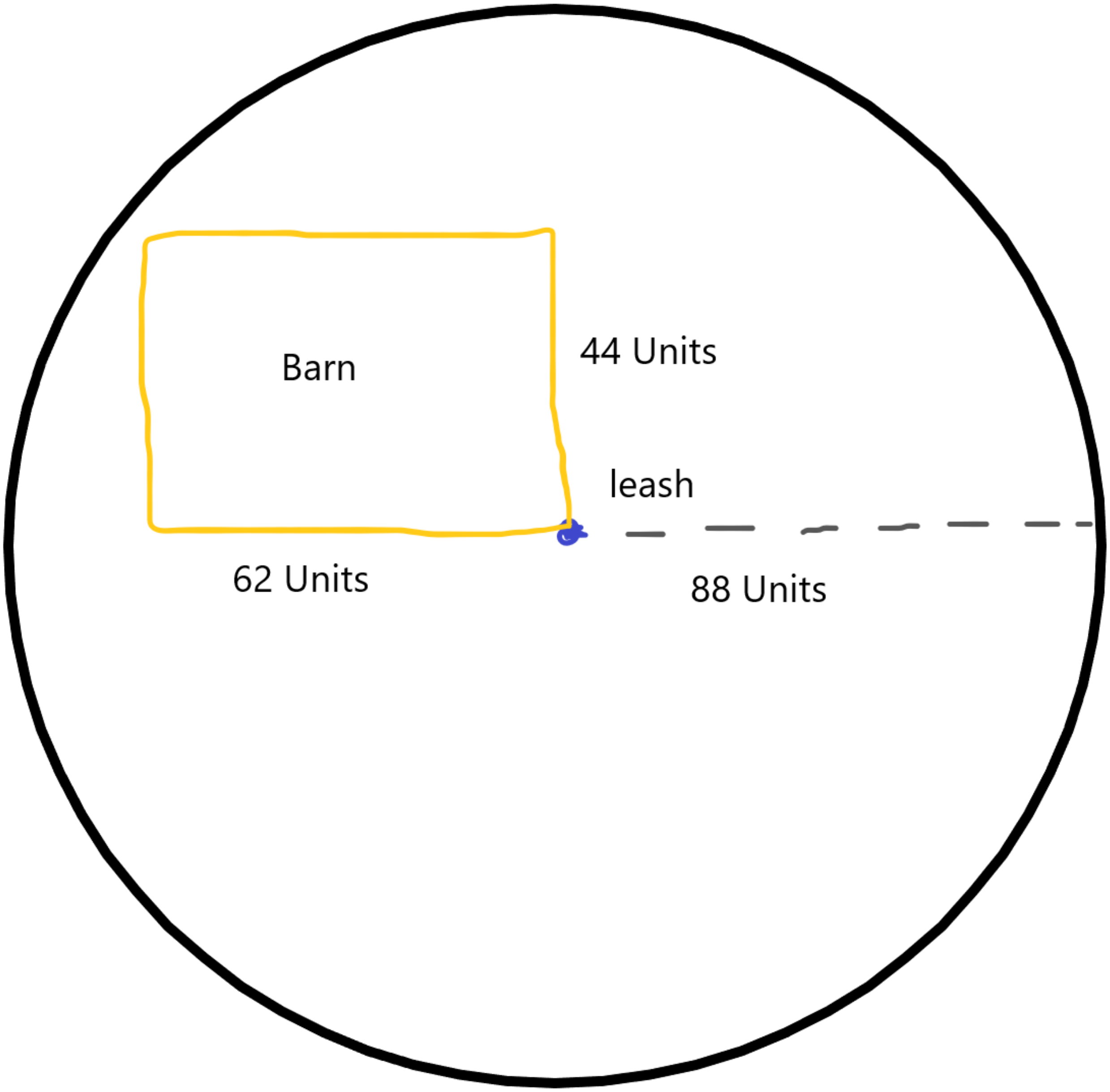
```
Type :h and hit Enter for context help.

[1]> (load "Assignment_02_Task_3.txt")
;; Loading file Assignment_02_Task_3.txt ...
;; Loaded file Assignment_02_Task_3.txt
T
[2]> die_edge
3.25
[3]> die_sides
6
[4]> die_side_area
10.5625
[5]> die_surface_area
63.375
[6]> dot_diameter
0.65
[7]> dot_nums
21
[8]> dot_radius
0.325
[9]> dot_area
0.33183068
[10]> dot_area_total
6.9684443
[11]> white_area
56.406555
[12]> percent_white
89.004425
[13]> (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?

Concept:

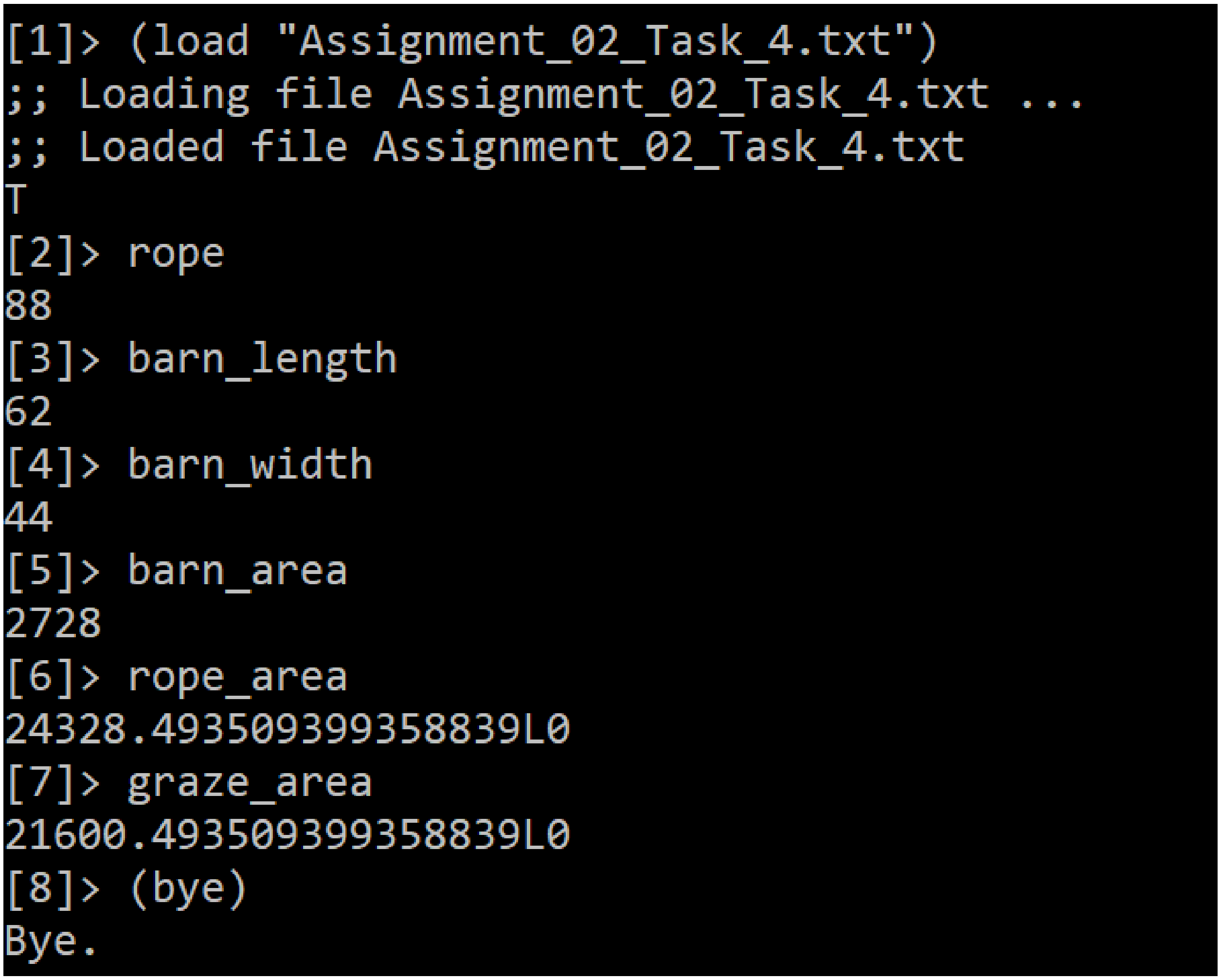


Code of “Tethered Goat”:

```
; Facts:
     (setf rope 88)
     (setf barn_length 62)
     (setf barn_width 44)

; Calculations:
     (setf barn_area (* barn_length barn_width))
     (setf rope_area (* pi rope rope))
     (setf graze_area (- rope_area barn_area))
```

Demo of “Tethered Goat”:



```
[1]> (load "Assignment_02_Task_4.txt")
;; Loading file Assignment_02_Task_4.txt ...
;; Loaded file Assignment_02_Task_4.txt
T
[2]> rope
88
[3]> barn_length
62
[4]> barn_width
44
[5]> barn_area
2728
[6]> rope_area
24328.493509399358839L0
[7]> graze_area
21600.493509399358839L0
[8]> (bye)
Bye.
```