COŞKUN HASAN ŞALTU

1801042631

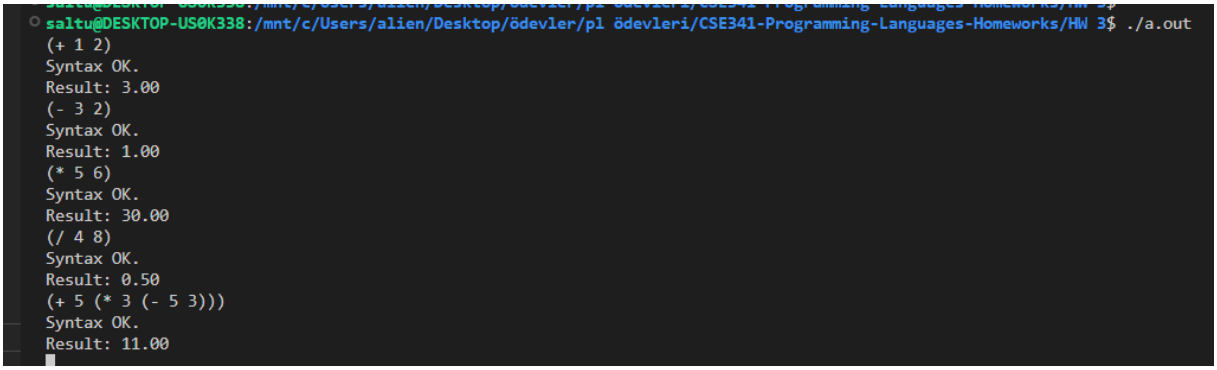
TEST CASE OF HW3

PART 1

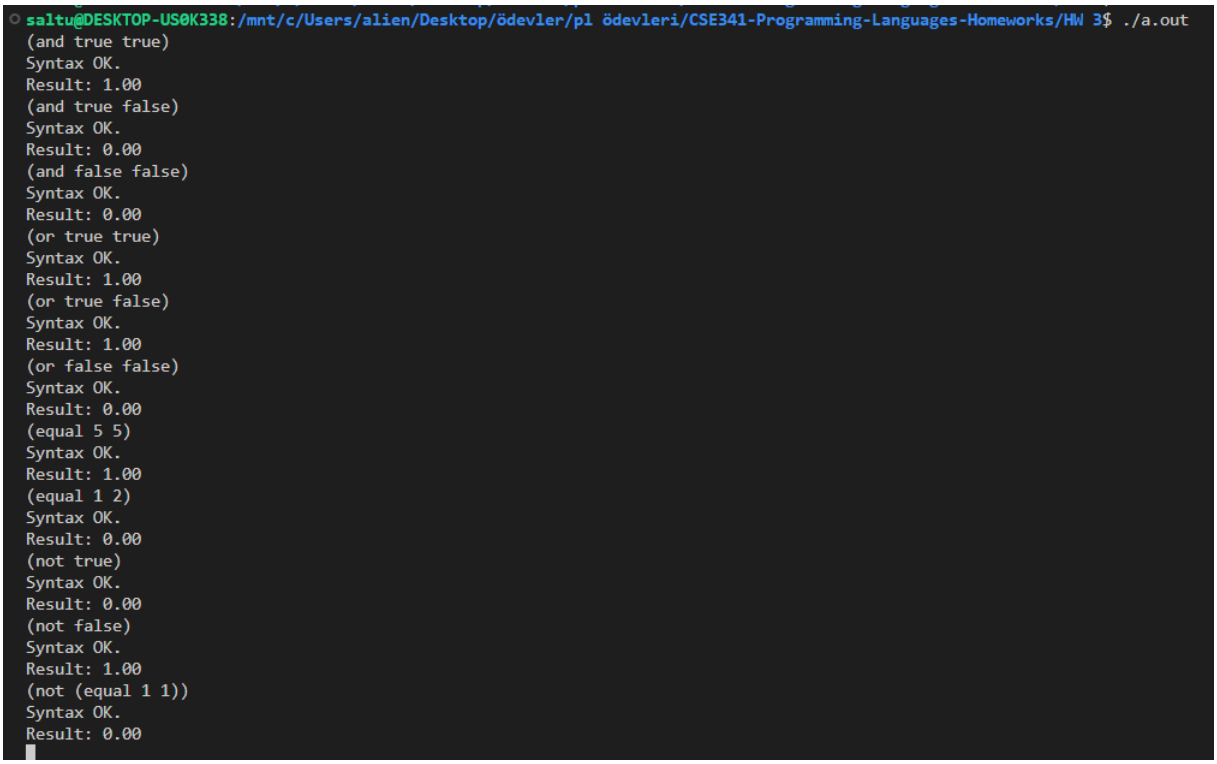
Part 1 contains makefile. Write the terminal “make” for compiling.

You can execute ./a.out file

+ - * / Test



and or equal not Test



Set Test

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ ./a.out
(set coskun 5)
Syntax OK.
Result: 5.00
(set hasan 10)
Syntax OK.
Result: 10.00
(set saltu 20)
Syntax OK.
Result: 20.00
(+ coskun hasan)
Syntax OK.
Result: 15.00
(* saltu (/ coskun hasan))
Syntax OK.
Result: 10.00
(+ ilkkan coskun)
Undefined variable ilkkan
```

İf Test

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ ./a.out
(set coskun 10)
Syntax OK.
Result: 10.00
(set hasan 5)
Syntax OK.
Result: 5.00
(if (equal coskun hasan) (*coskun hasan) (+ hasan coskun))
Syntax OK.
Result: 15.00
(if (not (equal coskun hasan)) (* coskun hasan) (+ hasan coskun))
Syntax OK.
Result: 50.00
```

Fraction check e.g. 5f3

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ ./a.out
(+ 5f2 3)
Syntax OK.
Result: 5.50
(* 1f3 3)
Syntax OK.
Result: 1.00
```

PART 2

You can execute the program write “clisp gpp_interpreter.lisp”.

+ - * / Test

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ clisp gpp_interpreter.lisp
Enter input string: (+ 1 2)

3.0
Enter input string: (* 2 3)

6.0
Enter input string: (/ 5 2)

2.5
Enter input string: (- 10 50)

-40.0
Enter input string: (+ 3 (* 2 (/ 5 2)))

8.0
```

and or equal not Test

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ clisp gpp_interpreter.lisp
Enter input string: (and true true)

"TRUE"
Enter input string: (and true false)

"FALSE"
Enter input string: (and false false)

"FALSE"
Enter input string: (or true true(

"SYNTAX ERROR"
Enter input string: (or true true)

"TRUE"
Enter input string: (or true false)

"TRUE"
Enter input string: (or false false)

"FALSE"
Enter input string: (equal 5 5)

"TRUE"
Enter input string: (not (equal 5 5))

"FALSE"
Enter input string: (and (equal 5 5) (equal 5 4))

"FALSE"
Enter input string:
```

Set and Disp Test

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ clisp gpp_interpreter.lisp
Enter input string: (set coskun 5)

"SYNTAX OK!"
Enter input string: (set hasan 10)

"SYNTAX OK!"
Enter input string: (set saltu 20)

"SYNTAX OK!"
Enter input string: (* coskun hasan)

50.0
Enter input string: (+ saltu coskun)

25.0
Enter input string: (* saltu (/ hasan saltu))

10.0
Enter input string: (set coskun 10)

"SYNTAX OK!"
Enter input string: (disp coskun)

10.0
Enter input string: (set coskun 5)

"SYNTAX OK!"
Enter input string: (disp coskun)

5.0
Enter input string: (disp 5)

5.0
Enter input string:
```

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ clisp gpp_interpreter.lisp
Enter input string: (+ selam 10)

"SYNTAX ERROR"
Enter input string: (set selam 20)

"SYNTAX OK!"
Enter input string: (+ selam 10)

30.0
Enter input string:
```

If Test

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ clisp gpp_interpreter.lisp
Enter input string: (set coskun 10)

"SYNTAX OK!"
Enter input string: (set hasan 5)

"SYNTAX OK!"
Enter input string: (if (equal coskun hasan) (* coskun hasan) (+ coskun hasan)

"SYNTAX ERROR"
Enter input string: (if (equal coskun hasan) (* coskun hasan) (+ coskun hasan))

15.0
Enter input string: (set hasan 10)

"SYNTAX OK!"
Enter input string: (if (equal coskun hasan) (* coskun hasan) (+ coskun hasan))

100.0
```

While Test

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ clisp gpp_interpreter.lisp
Enter input string: (set count 0)

"SYNTAX OK!"
Enter input string: (set mahmut 2)

"SYNTAX OK!"
Enter input string: (while (equal count 5) (set count (+ 1 count)) (set mahmut (* 2 mahmut)))

"SYNTAX OK!"
Enter input string: (disp count)

5.0
Enter input string: (disp mahmut)

64.0
```

```
saltu@DESKTOP-US0K338:/mnt/c/Users/alien/Desktop/ödevler/pl ödevleri/CSE341-Programming-Languages-Homeworks/HW 3$ clisp gpp_interpreter.lisp
Enter input string: (set count 10)

"SYNTAX OK!"
Enter input string: (while (equal count 0) (set count (- count 1)))

"SYNTAX OK!"
Enter input string: (disp count)

0.0
Enter input string:
```