```
user@cs3224:~$ cd schemeprogram/
user@cs3224:~/schemeprogram$ mit-scheme
MIT/GNU Scheme running under GNU/Linux
Type `^C' (control-C) followed by `H' to obtain information about interrupts.

Copyright (C) 2011 Massachusetts Institute of Technology
This is free software; see the source for copying conditions. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR
PURPOSE.

Image saved on Tuesday October 22, 2013 at 12:31:09 PM
  Release 9.1.1 || Microcode 15.3 || Runtime 15.7 || SF 4.41 || LIAR/i386 4.118 || Edwin 3.116

1 ]=> (load "schemeassignment.scm")

;Loading "schemeassignment.scm"... done
;Value: permutation

1 ]=> (define list1 '( 1 34 67 89 12 900 2))

;Value: list1

1 ]=> (define list2 '(a b c d e f))

;Value: list2

1 ]=> (reverseL list1)

;Value 11: (2 900 12 89 67 34 1)

1 ]=> (reverseL list2)

;Value 12: (f e d c b a)

1 ]=> (reverseL '(2 3 1 7 6))

;Value 13: (6 7 1 3 2)

1 ]=> (union '(1 2 5 6 7 8) '(1 2 5 6))

;Value 14: (8 7 1 2 5 6)

1 ]=> (union '(1 3 5 6 7) '(89 6 23))

;Value 15: (7 5 3 1 89 6 23)

1 ]=> (selectionSort '( 1 34 67 900 23 6 5 9))

;Value 16: (1 5 6 9 23 34 67 900)
```

```
1 ]=> (selectionSort '( 9 8 6 7 4 5 3 1 2))

;Value 17: (1 2 3 4 5 6 7 8 9)

1 ]=> (maxmin '(23 76 233 456 98 19999))

;Value 18: (19999 23)

1 ]=> (maxmin '(0 34 12 56 98))

;Value 19: (98 0)

1 ]=> (define list3 '( 1 2 3))

;Value: list3

1 ]=> (permutation list3)

;Value 20: ((1 2 3) (1 3 2) (2 1 3) (2 3 1) (3 2 1) (3 1 2))

1 ]=>
```

```
Last login: Wed Oct 11 20:06:09 on ttys000
ISHITAs-MacBook-Air:~ ishitaverma$ cd Desktop/
ISHITAs-MacBook-Air:Desktop ishitaverma$ cd haskellPgms/
ISHITAs-MacBook-Air:haskellPgms ishitaverma$ ghci
GHCi, version 8.2.1: http://www.haskell.org/ghc/  :? for help
Prelude> :set prompt "ghci>"
ghci>:load haskellassignment.hs
[1 of 1] Compiling Main             ( haskellassignment.hs, interpreted )
Ok, 1 module loaded.
ghci>reverseList ['a', 'b', 'c', 'd']
"dcba"
ghci>reverseList [1, 2, 4, 5, 6, 8]
[8,6,5,4,2,1]
ghci>unionL [1,2,3] [4,3,6,7]
[4,3,6,7,2,1]
ghci>unionL [34, 56, 90] [45, 88, 56, 67]
[45,88,56,67,90,34]
ghci>quicksort [1,4,56,7,2,8]
[1,2,4,7,8,56]
ghci>maxmin [1, 4, 56, 2, 45, 3]
[56,1]
ghci>maxmin [2, 0, 34, 12, 23, 567, 1000, 22]
[1000,0]
ghci>permutation [1,2,3]
[[1,2,3],[1,3,2],[2,1,3],[2,3,1],[3,1,2],[3,2,1]]
ghci>
```