Homework 2 - Lisp

Installation

SBCL

Steel **B**ank **C**ommon **L**ISP

A high performance **ANSI Common Lisp** implementation.

SBCL Official Website:
http://www.sbcl.org/index.html

Install SBCL on Windows

Go to this website,

https://github.com/akovalenko/sbcl-win32-threads/wiki

32 bit:
sbcl-1.1.4.0.mswin.1288-90ab477-x86.msi

64 bit:
sbcl-1.1.4.0.mswin.1288-90ab477-x86-64.msi

Install SBCL on Linux

use APT

```
$ sudo apt-get install sbcl
```

use YUM

```
$ sudo yum install sbcl
```

others

```
That's your business...XD
```

Install SBCL on OS X

use homebrew

```
$ brew install sbcl
```

use macports

```
$ sudo port install sbcl
```

If you never believe anyone,

Compile by yourself...

http://www.sbcl.org/getting.html

Command Line Interface

for Windows

命名提示字元

for Unix-like OS

終端機

Interactive Env

```
$ sbcl
```

Interactive Env

```
$ sbcl
This is SBCL 1.0.57.0.debian, an implementation of ANSI
Common Lisp.
More information about SBCL is available at
<http://www.sbcl.org/>.

SBCL is free software, provided as is, with absolutely no
warranty.
It is mostly in the public domain; some portions are
provided under
BSD-style licenses.  See the CREDITS and COPYING files in
the
distribution for more information.
*
```

```
* ( + 1 2 )
```

```
* ( + 1 2 )

3
```

```
* ( + 1 2 )


3
* ( + 1 2 3 )
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
* "Hello World"
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
* "Hello World"

"Hello World"
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
* "Hello World"

"Hello World"
* (DEFUN FIB(n)
   (IF (< n 2)
    n
     (+ (FIB (- n 1)) (FIB (- n 2)))))
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
* "Hello World"

"Hello World"
* (DEFUN FIB(n)
   (IF (< n 2)
    n
     (+ (FIB (- n 1)) (FIB (- n 2)))))

FIB
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
* "Hello World"

"Hello World"
* (DEFUN FIB(n)
   (IF (< n 2)
    n
     (+ (FIB (- n 1)) (FIB (- n 2)))))

FIB
* (FIB 20)
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
* "Hello World"

"Hello World"
* (DEFUN FIB(n)
   (IF (< n 2)
    n
     (+ (FIB (- n 1)) (FIB (- n 2)))))

FIB
* (FIB 20)

6765
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
* "Hello World"

"Hello World"
* (DEFUN FIB(n)
   (IF (< n 2)
    n
     (+ (FIB (- n 1)) (FIB (- n 2)))))

FIB
* (FIB 20)

6765
* (exit)
```

```
* ( + 1 2 )

3
* ( + 1 2 3 )

6
* "Hello World"

"Hello World"
* (DEFUN FIB(n)
   (IF (< n 2)
    n
     (+ (FIB (- n 1)) (FIB (- n 2)))))

FIB
* (FIB 20)

6765
* (exit)
$
```

Script File

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
```

```
;;; file: fib.lsp

(DEFUN FIB(n)
 (IF (< n 2)
  n
  (+ (FIB (- n 1)) (FIB (- n 2)))
 )
)

(print (FIB 20))
```

Execution

```
$ sbcl --script fib.lsp

6765
```

If you still have any question about SBCL,

Read **T**he **F**riendly **M**anual.

ANSI Common Lisp Tutorial:

http://acl.readthedocs.org/en/latest/

SBCL Manual:

http://www.sbcl.org/manual/index.html

Read **T**he **F**riendly **M**anual or
Use **T**he **F**riendly **G**oogle first,

before you ask teacher or TAs.

Merge Sort

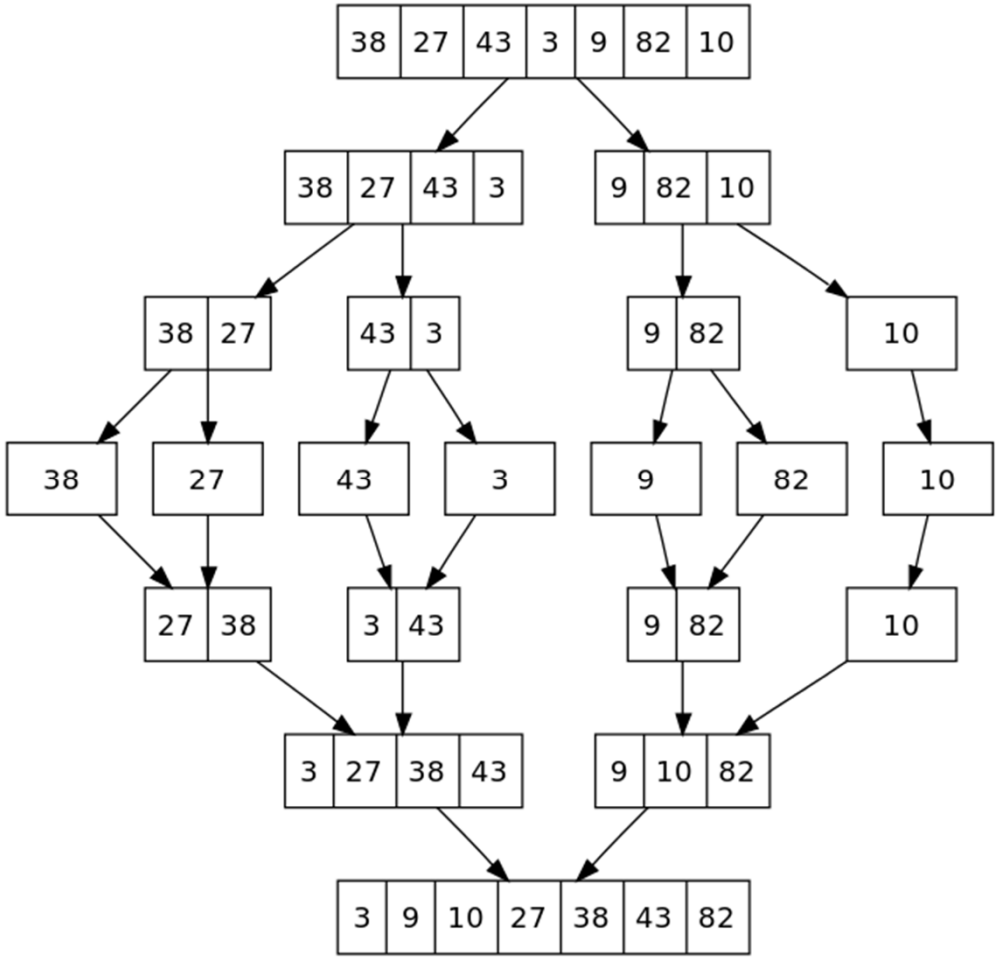

Photo from: Wikipedia

輸入格式

- 先輸入一個整數 N 表示數列中有多少數字
- 再輸入 N 個整數作為數列中的數字

範例輸入

```
Example 1:
3
3 2 1

Example 2:
5
1 3 8 9 1

Example 3:
10
9 8 16 2 7 199 0 98 1 29
```

輸出格式

- 輸出包含 N 個整數的已排序數列

範例輸出

```
Example 1:
1 2 3

Example 2:
1 1 3 8 9

Example 3:
0 1 2 7 8 9 16 29 98 199
```

參考程式碼

可基於以下程式碼進行修改

```
(defun mergesort (numbers)
 (return-from mergesort numbers))

; main function
(let ((n (read))
      (numbers))
 (setf numbers
  (do ((i 0 (+ i 1))
       (tmp nil))
    ((>= i n)
     (reverse tmp))
    (setf tmp (cons (read) tmp))))
 (format t "~{~A ~}~%" (mergesort numbers)))
```

Deadline

2015/04/26 22:00