

## Лабораторная работа №1

### Обработка списков в языке CLISP

Номер задачи выбирается по последней цифре пароля.

Присылаемый на проверку архив должен содержать 2 файла:

- файл отчета, содержащий титульный лист, условие задачи, исходный текст программы и результаты работы программы (можно в виде скриншотов);
- файл с исходным текстом программы на языке CLISP.

#### Задание на лабораторную работу

Напишите на языке CLISP программу для работы со списками по заданию. Обязательно использование рекурсии. Ввод всех входных данных должен запрашиваться с клавиатуры в процессе работы программы (функции). Не допускается использование: функционалов, а в теле рекурсивной функции - операторов SET и SETQ.

Номер варианта выбирается по последней цифре пароля.

0. Сформируйте список из позиций элемента, заданного параметром X, в списке L (нумерация элементов начинается с 1).  
Например, при X=4, L=(8 3 4 6 4 1) функция должна вернуть (3 5).
1. Преобразуйте одноуровневый список L во вложенный по следующему правилу.  
Например, при L=(a s d f g) функция должна вернуть (a (s (d (f (g))))).
2. Преобразуйте одноуровневый список L во вложенный по следующему правилу.  
Например, при L=(a s d f g) функция должна вернуть (((((a) s) d) f) g).
3. Переставьте элементы списка L таким образом, чтобы одинаковые элементы оказались рядом. Сортировку не использовать!  
Например, при L=(1 5 2 1 4 3 1 2 4 5 4) функция должна вернуть (1 1 1 5 5 2 2 4 4 4 3).
4. Сформируйте список из N элементов списка L, начиная с K-го элемента. (нумерация элементов должна начинаться с 1).  
Например, при L=(-2 6 s -1 4 f 0 z x r), K=3, N=4 функция должна вернуть (s -1 4 f).
5. Удалите элементы из первого списка L1 с номерами из второго списка L2 (второй список упорядочен по возрастанию, нумерация элементов должна начинаться с 1).  
Например, при L1=(a s d f g h j k l), L2=(1 4 5 8) функция должна вернуть (s d h j l).
6. Сформируйте список, состоящий из сумм первого и последнего, второго и предпоследнего элементов числового списка и т.д. Каждый элемент должен участвовать в сложении не более одного раза.

Например, при (1 -2 -3 4 5 6 -7 8 9) функция должна вернуть (10 6 -10 10 5).

7. Сформируйте список из пар соседних элементов списка L, одинаково отстоящих от начала и конца списка.

Например, при L=(1 a b c 2 3 f) функция должна вернуть ((1 f) (a 3) (b 2) (c))

8. Объедините 2 списка L1 и L2 в один, чередуя элементы списков.

Например, при L1=(1 2 3 4 5 6 7 8), L2=(a s d f) функция должна вернуть (1 a 2 s 3 d 4 f 5 6 7 8).

9. Добавьте элемент в конец числового списка L, чтобы сумма элементов нового списка стала равна 100.

Например, при L=(2 10 15 50 30) функция должна вернуть (2 10 15 50 30 -7).

### **Методические указания к выполнению лабораторной работы №1**

**Пример.** Объедините два упорядоченных по неубыванию списка в новый упорядоченный по неубыванию список без использования сортировки.

#### **Программа на языке CLISP**

```
(defun f()
  (princ "Vvedite pervii spisok ")
  (terpri)
  (setq L1 (read))
  (princ "Vvedite vtoroi spisok ")
  (terpri)
  (setq L2 (read))
  (princ "Rezult= ")
  (new L1 L2)
)
(defun new(L1 L2)
  (cond
    ((null L1) L2)
    ((null L2) L1)
    ((< (car L1) (car L2)) (cons (car L1) (new (cdr L1) L2)))
    (t (cons (car L2) (new L1 (cdr L2)))))
  )
)
```

Скриншот работы программы:

```
LispIDE -
File Edit Search View Settings Window Help
[Icons]
Document2 Document3
1 (defun f())
2   (princ "Uvedite pervii spisok ")
3   (terpri)
4   (setq L1 (read))
5   (princ "Uvedite vtoroi spisok ")
6   (terpri)
7   (setq L2 (read))
8   (princ "Rezult= ")
9   (new L1 L2)
10 )
11 (defun new(L1 L2)
12   (cond
13     ((null L1) L2)
14     ((null L2) L1)
15     ((< (car L1) (car L2)) (cons (car L1) (new (cdr L1) L2)))
16     (t (cons (car L2) (new L1 (cdr L2)))))
17 )
18 )
<
Copyright (c) Sam Steingold, Bruno Haible 2001-2010
Напечатайте :h и нажмите Ввод для получения справки.
[1]>
F
[2]>
NEW
[3]> (f)
Uvedite pervii spisok
(1 2 3 3 3 6 7 10)
Uvedite vtoroi spisok
(1 1 3 8 9 9)
Rezult=
(1 1 1 2 2 3 3 3 3 6 7 8 9 9 10)
[4]>
```