

Построение новых функций в среде Common Lisp

Примеры функций обработки
списков

Сортировка по убыванию. Метод прямого выбора

Алгоритм сортировки списка путем прямого выбора включает в себя следующие шаги:

1. В исходном списке $S1$ находится минимальный элемент Min и помещается в результирующий список $S2$.
2. Этот элемент удаляется из списка $S1$, и процедура поиска повторяется. С каждым шагом список $S1$ уменьшается.
3. Когда список $S1$ станет пустым, список $S2$ станет результирующим, упорядоченным по возрастанию списком.

Определение минимального элемента

```
(DEFUN MINLIST(X)
(COND
((NULL (CDR X)) (CAR X))
(T (MIN (CAR X) (MINLIST (CDR X)))))
)
)
```

Определение минимального элемента

```
CL-USER 1 > (DEFUN MINLIST(X)
(COND
  ((NULL (CDR X)) (CAR X))
  (T (MIN (CAR X) (MINLIST (CDR X)))))
)
)
MINLIST
```

```
CL-USER 2 > (minlist '(1 2 -3 4))
-3
```

```
CL-USER 3 >
```

Удаление элемента из списка

```
(DEFUN DELLIST(X L)
(COND
((= X (CAR L)) (CDR L))
(T (CONS (CAR L) (DELLIST X (CDR L)))))
)
)
```

Удаление элемента из списка

```
CL-USER 3 > (DEFUN DELLIST(X L)  
(COND  
  ((= X (CAR L)) (CDR L))  
  (T (CONS (CAR L) (DELLIST X (CDR L)))))  
)  
)  
DELLIST
```

```
CL-USER 4 > (DELLIST 5 '(1 2 3 4 5 6))  
(1 2 3 4 6)
```

Сортировка по убыванию

```
(DEFUN SORT1(L1 L2)
(COND
  ((EQ L1 NIL) L2)
  (T
   (SORT1 (DELLIST (MINLIST L1) L1) (CONS
    (MINLIST L1) L2))
  ))
)

(DEFUN SORTV1(M)
  (SORT1 M '())
)
```

Сортировка по убыванию

```
CL-USER 5 > (DEFUN SORT1(L1 L2)
```

```
(COND
```

```
((EQ L1 NIL) L2)
```

```
(T
```

```
(SORT1 (DELLIST (MINLIST L1) L1) (CONS (MINLIST L1) L2))
```

```
))
```

```
)
```

```
SORT1
```

```
CL-USER 6 > (DEFUN SORTV1(M)
```

```
(SORT1 M '())
```

```
)
```

```
SORTV1
```

```
CL-USER 7 > (sortv1 '(1 2 3 4))
```

```
(4 3 2 1)
```


Сортировка по возрастанию. Определение
максимального элемента

```
(DEFUN MAXLIST(X)
  (COND
    ((NULL (CDR X)) (CAR X))
    (T (MAX (CAR X) (MAXLIST (CDR X)))))
  )
)
```

Сортировка по возрастанию. Определение максимального элемента

```
CL-USER 8 > (DEFUN MAXLIST(X)
  (COND
    ((NULL (CDR X)) (CAR X))
    (T (MAX (CAR X) (MAXLIST (CDR X)))))
  )
)
```

MAXLIST

```
CL-USER 9 > (maxlist '(1 9 -5 5))
```

Сортировка по возрастанию. Удаление элемента

```
(DEFUN DELLIST(X L)
  (COND
    ((= X (CAR L)) (CDR L))
    (T (CONS (CAR L) (DELLIST X (CDR L)))))
  )
)
```

Сортировка по возрастанию. Метод прямого выбора

```
(DEFUN SORT2(L1 L2)
```

```
(COND
```

```
((EQ L1 NIL) L2)
```

```
(T
```

```
(SORT2 (DELLIST (MAXLIST L1) L1) (CONS (MAXLIST  
L1) L2))
```

```
))
```

```
)
```

```
(DEFUN SORTV2(M)
```

```
(SORT2 M '())
```

```
)
```

Сортировка по возрастанию. Метод прямого выбора

```
CL-USER 3 > (DEFUN SORT2(L1 L2)
```

```
(COND
```

```
((EQ L1 NIL) L2)
```

```
(T
```

```
(SORT2 (DELLIST (MAXLIST L1) L1) (CONS (MAXLIST L1) L2))
```

```
)))
```

```
SORT2
```

```
CL-USER 4 > (DEFUN SORTV2(M)
```

```
(SORT2 M '())
```

```
SORTV2
```

```
CL-USER 5 > (sortv2 '(2 -9 5 -6))
```

```
(-9 -6 2 5)
```

Пример лабораторной работы №3

Дан список целых чисел и целое число $a > 0$. Сформировать список элементов исходного списка, значения которых меньше a . Упорядочить полученный список по убыванию значений элементов.

Пример лабораторной работы №3

```
(DEFUN FORMLIST(X A)
  (COND
    ((NULL X) NIL)
    ((< (CAR X) A)
      (CONS (CAR X) (FORMLIST (CDR X) A)
    ))
    (T (FORMLIST (CDR X) A))
  )
)
```

Пример лабораторной работы №3

```
CL-USER 8 : 1 > (DEFUN FORMLIST(X A)
(COND
  ((NULL X) NIL)
  ((< (CAR X) A)
    (CONS (CAR X) (FORMLIST (CDR X) A)
  ))
  (T (FORMLIST (CDR X) A))
)
)
FORMLIST
```


Пример лабораторной работы №3

CL-USER 9 : 1 > (formlist '(2 4 5 -6 8) 3)

(2 -6)

CL-USER 10 : 1 > (formlist '(1 2 3 4 5 6) 4)

(1 2 3)

Пример лабораторной работы №3. Проверка списка

```
(DEFUN INTLIST(X)
  (COND ((NULL X) NIL)
        ((AND (= 1 (LENGTH X)) (INTEGERP (CAR X)))
         T)
        ((AND (INTEGERP (CAR X)) (INTLIST (CDR
X)))) T)
))
```

Пример лабораторной работы №3. Проверка списка

```
CL-USER 1 > (DEFUN INTLIST(X)
  (COND ((NULL X) NIL)
        ((AND (= 1 (LENGTH X)) (INTEGERP (CAR X)))
         T)
        ((AND (INTEGERP (CAR X)) (INTLIST (CDR
X)))) T)
  ))
```

INTLIST

```
CL-USER 2 > (intlist '(1 2 3 4))
```

T

```
CL-USER 3 > (intlist '(1.5 6.2 7.8))
```

NIL

Пример лабораторной работы №3 с проверкой списка

```
(DEFUN FORML(X A)
```

```
(COND
```

```
((AND (INTLIST X) (INTEGERP A))
```

```
(COND
```

```
((NULL X) NIL)
```

```
((< (CAR X) A)
```

```
(CONS (CAR X) (FORMLIST (CDR X) A)
```

```
))
```

```
(T (FORMLIST (CDR X) A))
```

```
))
```

```
(T NIL)))
```

Пример лабораторной работы №3 с проверкой списка

```
CL-USER 9 : 2 > (DEFUN FORML(X A)
(COND
  ((AND (INTLIST X) (INTEGERP A))
    (COND
      ((NULL X) NIL)
      ((< (CAR X) A)
        (CONS (CAR X) (FORMLIST (CDR X) A)
      ))
      (T (FORMLIST (CDR X) A))
    ))
  (T NIL)
)
)
FORML
```

Пример лабораторной работы №3 с проверкой списка

CL-USER 10 : 2 > (form1 '(-1 -2 3 4 5) 0)

(-1 -2)

Пример лабораторной работы №3 с проверкой списка и сортировкой

(DEFUN lab3(X A)

(sortv1 (forml X A))

Пример лабораторной работы №3 с проверкой списка и сортировкой

```
CL-USER 15 : 2 > (DEFUN lab3(X A)
(sortv1 (form1 X A)))
```

LAB3

```
CL-USER 16 : 2 > (lab3 '(-2 -1 3 1 4 5) 4)
(3 1 -1 -2)
```