

Действия над элементами списка

Версия Maxima

Maxima 5.44.0



Работа со списками в Maxima

Выполнила:

Леонтьева Анна, ИВТ 1 курс

sum(выражение, i , a , b) -
суммирует выражения при
изменении i от a до b .

$sum(x/i, i, 1, 3) - x + x/2 + x/3 + x/4$

map(функция, выражение1, выраже-
ние2,...) -

применяет функцию (оператор,
операцию) к

последовательности выражений.

$map("=", [a, b], [1, -1]) - [a=1, b=-1]$

apply(функция, список) -

применяет функцию к
заданному списку (преобразует
список в аргументы функции).

$apply(max, [1, 5, 20]) - 20$

Основные функции

```
→ a1:[2,4,6,8];  
(1o1) [2,4,6,8]  
  
→ a2:[10,12,14,16];  
(1o2) [10,12,14,16]  
  
→ a3:[a1,a2,18,20,22];  
(1o3) [[2,4,6,8],[10,12,14,16],18,20,22]  
  
→ a3[2];  
(1o4) [10,12,14,16]  
  
→ a3[4];  
(1o5) 20
```

Рассмотрим основные функции для работы со списками.

length(список) - определяет количество элементов (длину).

length(list) - 3

copylist(список) - создаёт копию.

append(список1, список2, ...) - склеивает списки.

reverse(список) - меняет порядок следования элементов на обратный.

reverse(list) - [[x,y],a,1]

member(элемент,список) - возвращает *true*, если элемент входит в список, иначе *false*.

member(a,list) - true

Аналоги **Lisp**-а:)

first(список) - первый элемент ("car").

first(list) - 1

rest(список) - хвост, т.е. всё кроме первого элемента ("cdr").

rest(list) - [a,[x,y]]

last(список) - последний элемент.

cons(элемент,список) - добавить элемент в начало (создаёт новый список).

cons(t,list) - [t,1,a,[x,y]]

endcons(элемент,список) - добавить элемент в конец (создаёт новый список).

```
→ apply(max,a5);  
(1o6) 16  
  
→ apply(min,a5);  
(1o7) 2  
  
→ a6: makelist(3*x,x,a1);  
(1o8) [6,12,18,24]  
  
→ mes1: cons(6,a1);  
(1o9) [6,2,4,6,8]  
  
→ mes2: endcons(6,a1);  
(1o10) [2,4,6,8,6]  
  
→ reverse(a5);  
(1o11) [16,8,14,6,12,4,10,2]
```

Списки в Maxima

В наследство от функционального программирования, и **Lisp** в частности, **Maxima** получила списки как базовую форму представления данных. Списком является группа элементов, разделённых запятыми и заключённых в квадратные скобки (совсем как в **Python**). Список может быть пустым или содержать любые другие элементы, в том числе вложенные списки. Запись

list:[1,a,[x,y]] определяет список *list*, состоящий из 3-х элементов, последний из которых сам является списком. Обратиться к определённому элементу можно по индексу (начиная с 1):

list[2] - a
list[3] - [x,y]
list[3][1] - x

Генераторы списков

makelist(выражение, i, a, b) - создание списка на основе заданного выражения при изменении переменной *i* от *a* до *b*.

makelist(a/i, i, 1, 4) - [a, a/2, a/3, a/4]

makelist(выражение, x, список) - создание списка на основе заданного выражения из исходного списка.

makelist(a/x, x, [1,2,3,4]) - [a, a/2, a/3, a/4]

create_list(выражение, x1, список1, x2, список2, ...) - создание списка на основе заданного выражения из нескольких исходных списков.