ЛАБОРАТОРНАЯ РАБОТА №1

БАЗОВЫЕ ФУНКЦИИ ЯЗЫКА ЛИСП

1.1. Цель работы

Целью работы является изучение базовых функций

языка Лисп, предназначенных для организации и обработки

списков.

1.2.Базовые функции Лиспа

Базовые функции обработки можно сравнить с основными

арифметическими действиями: они просты и их мало.

Назначение этих функций состоит в:

-расчленении S-выражений (селекторы);

-составлении S-выражений (конструкторы);

-анализ S-выражений (предикаты).

Таблица базисных выражений

|  |  |  |
| --- | --- | --- |
| Назначение | Вызов | Результат |
| селектор | (CAR список ) | S-выражение |
| селектор | (CDR список ) | список |
| конструктор | (CONS S-выраж.,список ) | список |
| предикат | (ATOM S-выражение) | T,NIL |
| предикат | (EQ атом атом ) | T,NIL |
|  |  |  |

1.3.1.Функция CAR. Значением функции является голова списка-аргумента

Примеры:

(CAR '(P Q R))-результат P;

(CAR '(A B))- результат A;

(CAR '(DOG CAT))-результат DOG;

(CAR '((A B) C))-результат (A B);

(CAR (A B))-результат не определен, т.к. (A B)-рассматривается как функция.

1.3.2.Функция CDR. Значением функции является хвост списка-аргумента.

Примеры:

(CDR '(P Q R))-результат (Q R);

(CDR '(B))-результат NIL;

В ЛИСпе наряду с базовыми функциями CAR и CDR определены функции, выделяющие любой из десяти первых элементов: FIRST- выделяет первый элемент списка, SECOND- второй и т.д.

1.3.3. Функция CONS

Функция CONS имеет два аргумента - первый S-выражение , вторым аргументом обязательно должен быть список. Назначение функции CONS-конструирование нового списка. Новый список получаем путем добавления первого аргумента к списку- второму аргументу.

Для получения списка, содержащего один элемент, нужно в качестве второго списка указать пустой список NIL.

Примеры:

(CONS 'A '(B C))-результат список (A B C );

(CONS '(FIRST SECOND)'(THIRD FOURTH FIFTH))- результат

((FIRST SECOND) THIRD FOURTH FIFTH);

(CONS NIL NIL) - результат пустой список (NIL);

(CONS 12 NIL) результат список ( 12 )

(CONS '(A B C) NIL) - список ((A B C ));

1.3.4. Связь между функциями CAR, CDR, CONS.

Список, разделенный с помощью CAR и CDR на голову и хвост, можно объединить с помощью функции-конструктора CONS.

1.3.5.Предикат ATOM

Предикат- это логическая функция. Предикат ATOM возвращает значение T ( эквивалент TRUE), если его аргументом является атом и NIL (эквивалент FALSE), иначе.

Примеры использования ATOM:

(ATOM 'X) - результат T, т.к. аргумент-атом;

(ATOM '(1 2 3)) - результат NIL, аргумент-список;

(ATOM (CDR'(1 2 3))) - результат NIL;

(ATOM (CAR'(A B C))) – результат T

1.3.6.Предикат EQUAL

Предикат EQUAL проверяет эквивалентность любых S-выражений.

Примеры :

(EQUAL '(3 3)'(4 3)) - NIL, так как списки не совпадают

(EQUAL 'Y 'Y) - T, атомы совпадают;

(EQL 5.0 5.0) - T

1.3.7.Расширение базовых функций

Кроме базовых функций есть целый набор встроенных функций, выполняющих преобразования S-выражений и дополняющих базовые функции.

1.3.7.1.Вложенные вызовы функций CAR и CDR

Используя функции CAR и CDR можно выделить любой элемент

списка. Например, для выделения второго элемента второго подсписка нужно записать:

(CAR(CDR(CAR(CDR '((A B C) (D E) (F H))))))

Для таких комбинаций CDR CAR можно использовать более короткую запись в виде одного вызова функции:

(C...R список ), где вместо многоточия записывается нужная комбинация букв A (CDR) и D (CDR).НО НЕ БОЛЕЕ 4 БУКВ.

Например, записанная выше конструкция может быть представлена:

(CADADR '((A B C) (D E) (F H )))

Для выделения из списка какого-либо элемента: первого, второго, десятого можно использовать специальные функции: FIRST, SECOND, THIRD, FOURTH, FIFTH, SEVENTH, EIGHTH, NINTH, TENTH. Все эти функции имеют один аргумент - список.

Более того, есть еще две функции, выделяющие элемент из списка: (NTH n, список)-выделяет n-ый элемент из списка второго аргумента функции NTH;

(LAST список) - выделяющая последний элемент из списка аргумента.

Примеры использования LAST, NTH:

(NTH 4 '(F D C C F H)) - выделяет элемент F, т.к. номер элементов начинается с нуля.

(LAST '(A B C D E F)) - выделяем последний элемент F.

1.3.7.2.Дополнительные предикаты сравнения аргументов

1.Предикат EQ

Предикат EQ сравнивает два атома, и принимает значение T, если атомы идентичны и NIL в противном случае. EQ сравнивает только атомы и константы T, NIL

Примеры :

(EQ 'X 'Y) - атомы неэквивалентны NIL;

(EQ () NIL) - T;

(EQ T (ATOM 'CAT)) - T

2.Предикат "="

Предикат "=" применяется для сравнения чисел различных типов. Предикат принимает значение T, если значение чисел совпадают вне зависимости от типа, иначе NIL.

Примеры :

(= 3 3) - T;

(= 3 7) - NIL

1.3.7.3.Предикат NUMBERP.

Предикат NUMBERP имеет один аргумент, который является S-выражением. Предикат принимает значение T, если аргумент является числом любого типа, и NIL, в любом другом случае.

Примеры :

(NUMBERP 3.066) - T;

(NUMBERP T) - NIL т.к. T-не число.

1.3.7.4.Предикат NULL.

NULL имеет один аргумент - список, если список пустой, то NULL принимает значение T, иначе NIL.

1.3.7.5.Функция LIST.

Эта функция может иметь любое количество аргументов, эта функция формирует список из аргументов .

Примеры :

(LIST '1 '2) - результат (1 2);

(LIST 'A 'B 'C 'D (+ 3 4)) - результат (A B C D 7).

1.4.Задание на лабораторную работу:

1. Ознакомиться с описанием лабораторной работы
2. Знакомиться со средой программирования LispWorks
3. Выполнить свой вариант задания, вариант выдает преподаватель. Задание выполнить различными способами, применяя базовые и дополнительные функции LispWorks

Задание 1:

Объединить хвосты трех списков в один список, исходные списки взять из таб.1.

Задание 2:

Создать новый список из элементов нескольких исходных списков. В качестве исходных списков использовать списки табл.1. Какие именно элементы этих списков нужно использовать, указано в таб.2.

Задание 3:

Проверить, являются ли числами элементы списка, полученного в результате выполнения задания 2, проверить совпадение элементов.

Задание 4: Запишите последовательности вызовов CAR и CDR, выделяющие из приведенных ниже списков символ «а». Упростите эти вызовы с помощью функций C...R.

а) (1 2 3 а 4)

б) (1 2 3 4 а)

в) ((1) (2 3) (а 4))

г) ((1) ((2 3 а) (4)))

д) ((1) ((2 3 а 4)))

е) (1 (2 ((3 4 (5 (6 а))))))

Задание 5: 5.1 Составьте список студентов своей группы

(ФИО ФИО ... ФИО)

Для каждого студента

а) с помощью функции LIST составьте следующие списки:

Для самого студента - (дата рождения), (адрес), (средний бал по лекционным занятиям), (средний бал по практическим занятиям), (средний бал по лабораторным работам). Для отца и матери - (ФИО), (дата рождения), (адрес), (место работы).

б) с помощью функций CONS и SETQ объедините полученные списки и присвойте их в виде значений символам, означающим ФИО каждого студента:

ФИО ст. - (((дата рождения ст.) (адрес ст.)((ср. бал(до десятых) по лекционным занятиям) (ср. бал по практическим занятиям) (ср. бал по лабораторным работам))) (((ФИО отца) (дата рождения отца) (адрес) (место работы отца)) ((ФИО матери) (дата рождения матери) (адрес) (место работы матери)))).

Все результаты должны быть прокомментированы: почему получен именно такой результат

1.5.Отчет по лабораторной работе

Отчет по лабораторной работе должен содержать:

- перечень и назначение базовых функций LISP;

- задание на лабораторную работу;

- протоколы работы в LispWorks

ТАБЛИЦА 1 Исходные списки

|  |  |  |  |
| --- | --- | --- | --- |
| Вариант | 1 | 2 | 3 |
| 1 | (Y U I) | (G1 G2 G3) | (KK LL MM JJJ) |
| 2 | (G55 G66 G777) | (9 (F G) I) | (N I L T D J (II JJ)) |
| 3 | ((PI) V (H J K)) | (R YU (H KJ KL) | (U II OO LL PP (3 4 5)) |
| 4 | (T (U U1 U2)(U4 U6 U8)) | (4 6 (7 8 9)) | (78 89 90 67 45) |
| 5 | (9 (() 8 88 888)) | (H (J K L) (UJN)) | (C B (N M I) (T Y U)) |
| 6 | (T Y D E F (NL KM LM) JL) | (+ 2 3) | (\*(+ 6 8) (- 70 8)) |
| 7 | (TYPE PRINT DEL) | (H (H J O) (UJ N)) | (READ SAVE LOAD (TXT)) |
| 8 | (GOAL FUNCTOR CLAUSE (DATA BASE)) | (2 5(5 4 6)8) | (L(K(K I)U)) |
| 9 | (DOG(CAT)FOX()) | (RET GET PUT OUT IN) | (MOV ADD (MUL DEV)) |
| 10 | (FIR SED(1 2 3) (5) ()) | (H J U K(L M N) (D E L)) | (4 5(6 7)) |
| 11 | (PRIM SD FLAG () (GHG)) | (1 56 98 52) | (T Y H) |
| 12 | (H G (U J) (T R)) | (2 1 (+ 4 5)) | (TYPE CHAR REAL (H G)) |
| 13 | (REM FFG HHJ(H J G D)) | (2 34 56 78(7 8)) | (UN Y LOOP) |
| 14 | (T (HJ(JH KL)) K) | (67 54 (8 9 0)(4 6)) | (K F G H) |
| 15 | (3 (3 4 5 Y U)((T Y)) | (G H (6 7 8)8 9 0 7 6) | ((5 T 7 Y H)U) |
| 16 | (Z X C S A D F) | ((R)(30)(3) 23)) | (U I 8 9 6 5 4 3 (1 2 3)) |
| 17 | (V B N J H ) | ((Y U I)(H J K)(8)78) | (df FG HJ K L (O 0 9)) |
| 18 | (D F G H J K) | (1 2 3 4 5 6 (4 5) 4) | (ER RT TY 5 6 6 5) |

ТАБЛИЦА 2

|  |  |  |  |
| --- | --- | --- | --- |
| Вариант | СПИСКИ | | |
| 1 | 2 | 2 | 3 |
| 2 | 3 | 2 | 6 |
| 3 | 1 | 3 | 6 |
| 4 | 2 | 3 | 4 |
| 5 | 2 | 3 | 3 |
| 6 | 6 | 2 | 2 |
| 7 | 3 | 2 | 3 |
| 8 | 4 | 3 | 2 |
| 9 | 3 | 4 | 3 |
| 10 | 4 | 4 | 3 |
| 11 | 5 | 3 | 2 |
| 12 | 3 | 3 | 3 |
| 13 | 4 | 4 | 2 |
| 14 | 2 | 3 | 3 |
| 15 | 2 | 3 | 2 |
| 16 | 5 | 4 | 7 |
| 17 | 3 | 4 | 6 |
| 18 | 6 | 6 | 6 |