|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

*ФАКУЛЬТЕТ «Информатика и системы управления»*

*КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»*

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 01 |

**Дисциплина:  *Функциональное и логическое программирование***

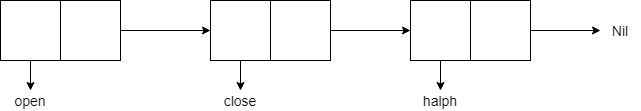
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ***ИУ7И-66Б*** |  |  | **Нгуен Ф. С.** |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | |  | | --- | | **Толпинская Н. Б.** | |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

*Москва, 2021*

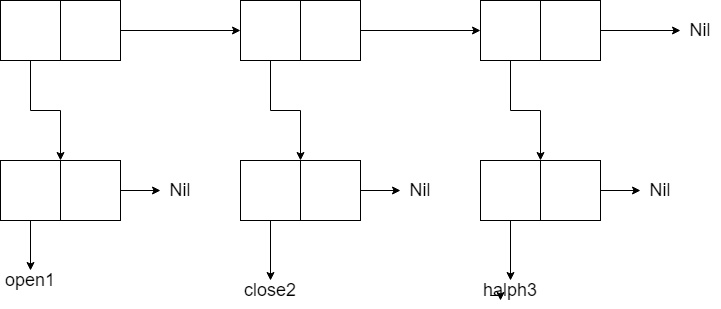
1. **Задание 1:**

Представить следующие списки в виде списочных ячеек:

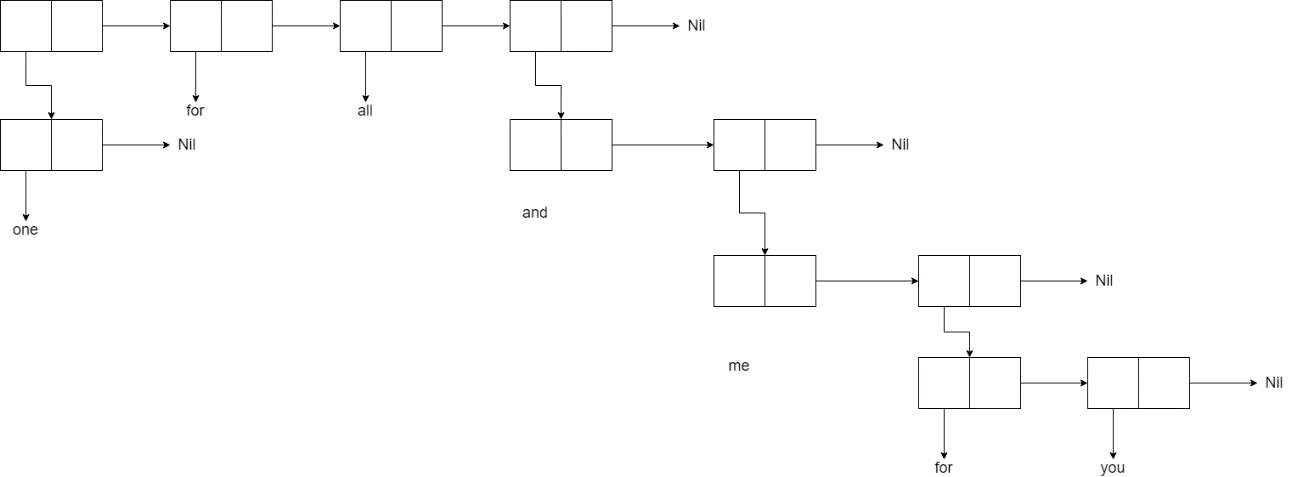
1. **’(open close halph)**
2. **’((open1) (close2) (halph3))**
3. **’((one) for all (and(me(for you))))**
4. **’((TOOL) (call))**
5. **’((TOOL1) ((call2)) ((sell)))**
6. **’(((TOOL) (call)) ((sell)))**
7. **’(open close halph)**

****

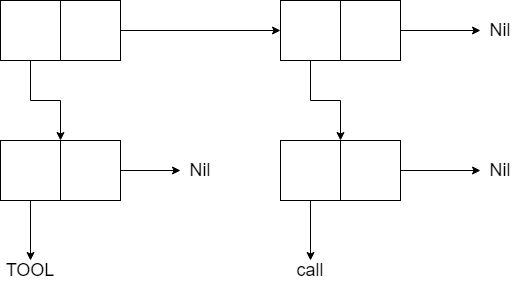
1. **’((open1) (close2) (halph3))**

****

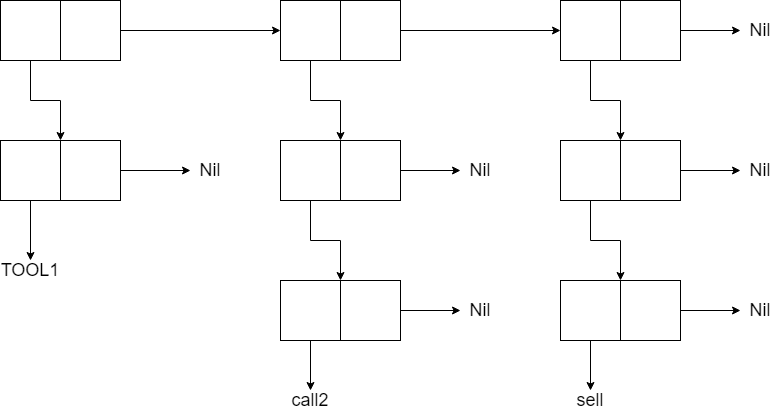
1. **’((one) for all (and(me(for you))))**

****

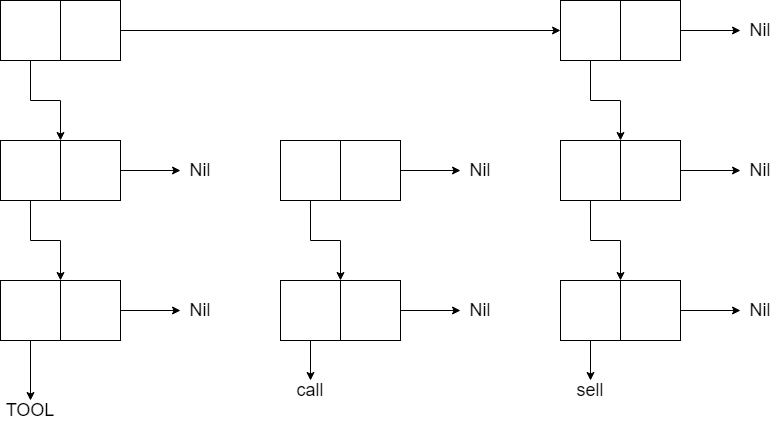
1. **’((TOOL) (call))**

****

1. **’((TOOL1) ((call2)) ((sell)))**

****

1. **’(((TOOL) (call)) ((sell)))**

****

1. **Задание 2:**

Используя только функции car и cdr, написать выражения, возвращающие:

1. *Второй элементы заданного списка:*

* (car (cdr ‘(1 2 3 4)))
* cadr

1. *Третий элементы заданного списка:*

* Caddr
* (car (cdr (cdr ‘(1 2 3 4))))

1. *Четвертый элементы заданного списка:*

* Cadddr
* (car (cdr (cdr (cdr ‘(1 2 3 4)))))

1. **Ответы на вопросы**
   1. *Элементы языка:*
      * **Атомы:** 
        + Символы;
        + Спесиальные символы;
        + Самоопределимые атомы;
      * **Точечные пары** ::= (<атом>.<атом>) | (<атом>.<точечная пара >) | (<точечная пара >.<атом>) | (<точечная пара >.< точечная пара >)
      * **Списки** ::= <пустой список> | <непустой список >
        + < пустой список > ::= () | Nil
        + < непустой список > ::= (<первый элемент >.< хвост >)
        + < первый элемент> ::= < S-выражениe >
        + <Хвост> ::= < список >
   2. *Синтаксис элемента языка и их представление в памяти:*

* Любая структура (точечная пара или список) заключается в круглые скобки (А. В)-точечная пара, (А) - список из одного элемента
* Пустой список изображается как Nil или ();
* Непустой список по определению может быть изображен:

(А. (В.(С.(D O)))), допустимо изображение списка последовательностью атомов, разделенных пробелами- (А В С D).

* Элементы списка могут, в свою очередь, быть списками (любой список заключается в круглые скобки), например (А (В С) (D (E))).
* Таким образом, синтаксически наличие скобок является признаком структуры- - списка или точечной пары.
* Любая непустая структура Lisp в памяти представляется списковой ячейкой, хранящей два указателя на голову (первый элемент) и хвост-все остальное.
  1. *Как воспринимается '*
     + ’(A B C D) аналог quote (блокирует вычисления - предохраняет свой единственный аргумент от вычисления), A B C D тогда воспринимаются как данные, а не как фрагмент программы
  2. *Что такое рекурсия?*
     + Рекурсия — это ссылка на определяемый объект во время его определения