

Московский государственный технический университет  
имени Н. Э. Баумана



Факультет: Информатика и системы управления

Кафедра: Программное обеспечение ЭВМ и информационные технологии

Дисциплина: Функциональное и логическое программирование

## Лабораторная работа №1

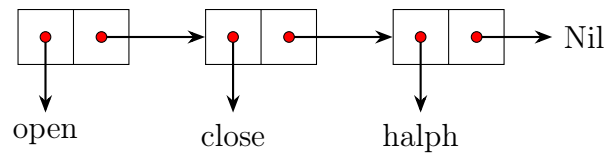
Выполнили: Никичкин А. С., Фокеев А. И.

Группа: ИУ7–61

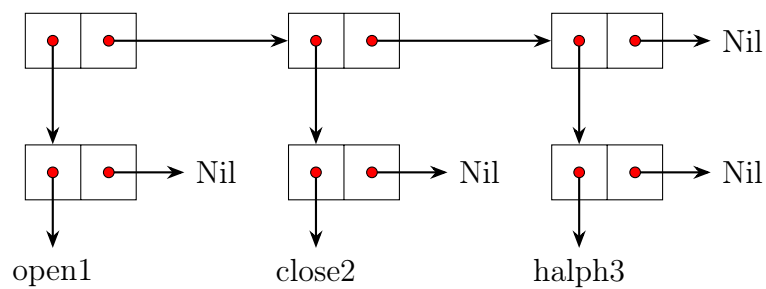
Москва, 2015 г.

# 1 Представить списки в виде списочных ячеек

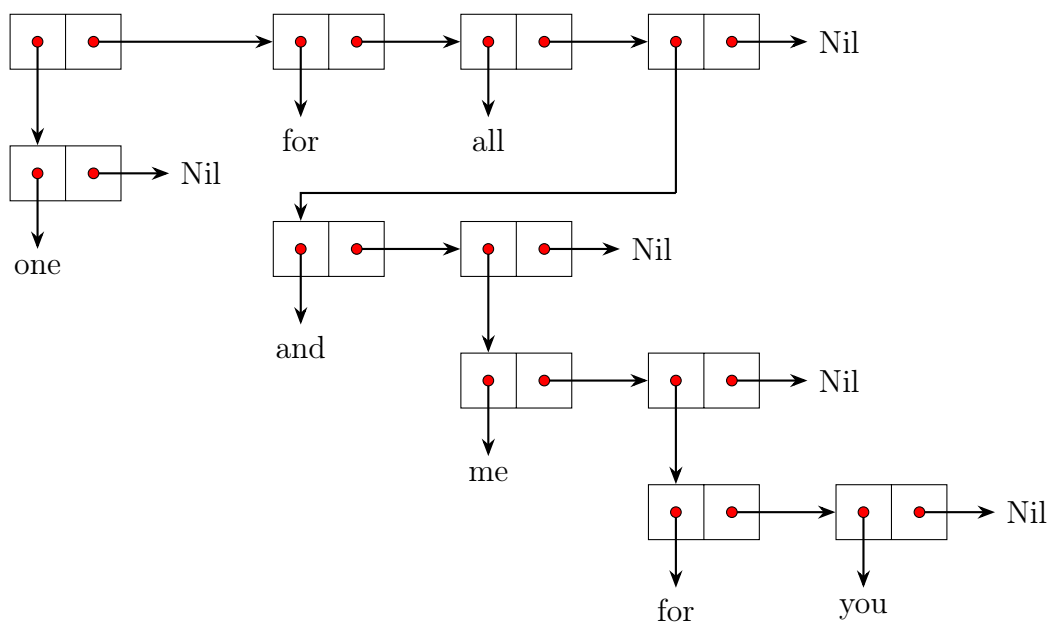
Задание 1.1  $'(open\ close\ halph)$



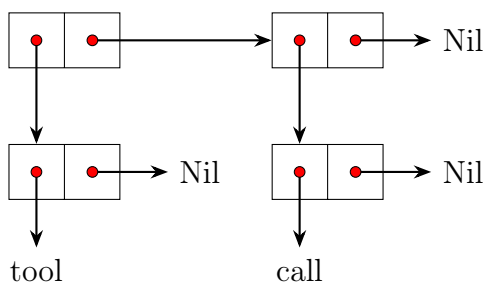
Задание 1.2  $'((open1)\ (close2)\ (halph3))$



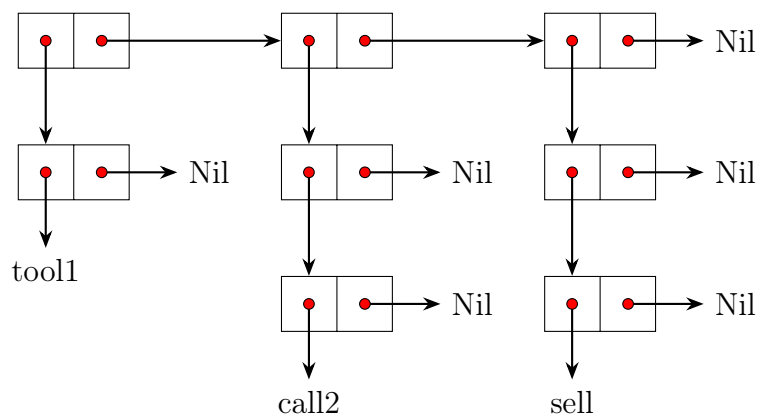
Задание 1.3  $'\left((one)\ for\ all\ \left(and\ (me\ (for\ you))\right)\right)$



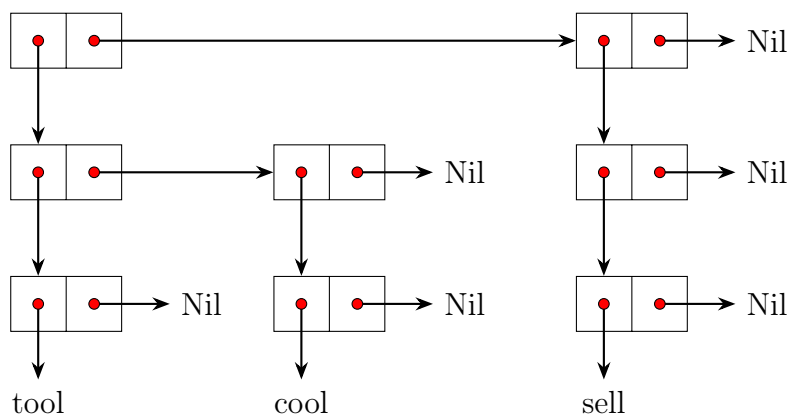
Задание 1.4  $'((tool) (call))$



Задание 1.5  $'((tool1) ((call2)) ((sell)))$



Задание 1.6  $'(((tool) (call)) ((sell)))$



## 2 Используя только функции CAR и CDR, написать выражение

**Задание 2.1** Возвращающие второй элемент списка

```
(car (cdr example-list)) = (cadr example-list)
```

**Задание 2.2** Возвращающие третий элемент списка

```
(car (cdr (cdr example-list))) = (caddr example-list)
```

**Задание 2.3** Возвращающие четвёртый элемент списка

```
(caddr example-list)
```

## 3 Что будет в результате вычисления выражения

**Задание 3.1**

```
(caadr '((blue cube) (red pyramid))) => RED
```

**Задание 3.2**

```
(cдар '((abc) (def) (ghi))) => NIL
```

**Задание 3.3**

```
(cadr '((abc) (def) (ghi))) => (DEF)
```

**Задание 3.4**

```
(caddr '((abc) (def) (ghi))) => (GHI)
```

## 4 Напишите результат вычисления выражения

**Задание 4.1**

```
(list 'Fred 'and 'Wilma) => (FRED AND WILMA)
```

**Задание 4.2**

```
(list 'Fred '(and Wilma)) => (FRED (AND WILMA))
```

**Задание 4.3**

```
(cons Nil Nil) => (NIL)
```

**Задание 4.4**

```
(cons T Nil) => (T)
```

**Задание 4.5**

```
(cons Nil T) => (NIL . T)
```

**Задание 4.6**

```
(list Nil) => (NIL)
```

**Задание 4.7**

```
(cons '(T) Nil) => ((T))
```

**Задание 4.8**

```
(list '(one two) '(free temp)) => ((ONE TWO) (FREE TEMP))
```

**Задание 4.9**

```
(cons 'Fred '(and Wilma)) => (FRED AND WILMA)
```

**Задание 4.10**

```
(cons 'Fred '(Wilma)) => (FRED WILMA)
```

**Задание 4.11**

```
(list Nil Nil) => (NIL NIL)
```

**Задание 4.12**

```
(list T Nil) => (T NIL)
```

**Задание 4.13**

```
(list Nil T) => (NIL T)
```

**Задание 4.14**

```
(cons T (list Nil)) => (T NIL)
```

**Задание 4.15**

```
(list '(T) Nil) => ((T) NIL)
```

**Задание 4.16**

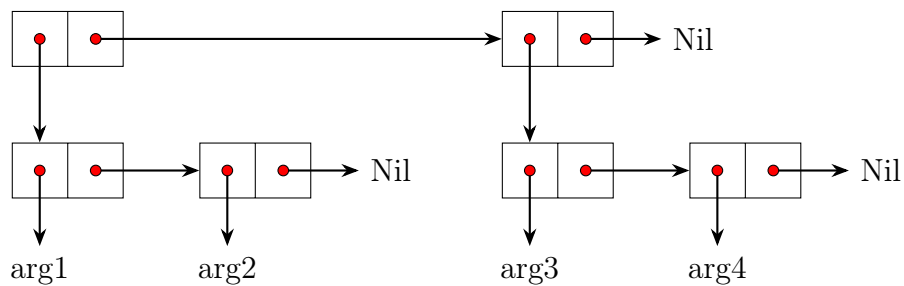
```
(cons '(one two) '(free temp)) => ((ONE TWO) FREE TEMP)
```

## 5 Написать функции и представить результаты в виде cons-ячеек

**Задание 5.1** Функция от четырёх аргументов возвращающая список вида

$$((arg1\ arg2)\ (arg3\ arg4))$$

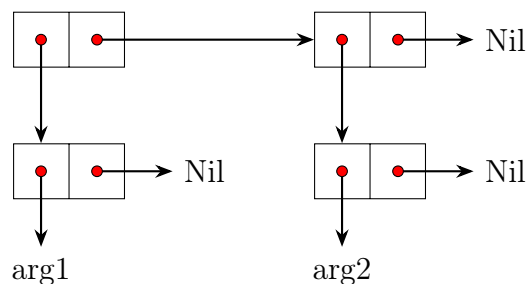
```
1 (defun problem-5-1 (arg1 arg2 arg3 arg4)
2   "Make list like ((arg1 arg2) (arg3 arg4))"
3   (list
4     (list arg1 arg2)
5     (list arg3 arg4)))
```



**Задание 5.2** Функция от двух аргументов возвращающая список вида

$$((arg1)\ (arg2))$$

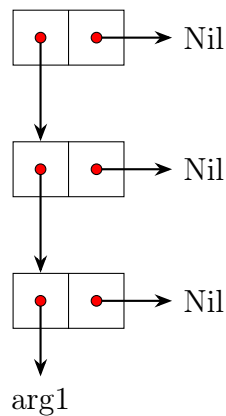
```
1 (defun problem-5-2 (arg1 arg2)
2   "Make list like ((arg1) (arg2))"
3   (list
4     (list arg1)
5     (list arg2)))
```



**Задание 5.3** Функция от одного аргумента возвращающая список вида

$((arg1))$

```
1 (defun problem-5-3 (arg1)
2   "Make list like (((arg1)))"
3   (list
4     (list
5       (list arg1))))
```



## 6 Вывод

В данной лабораторной работе было изучено понятие *cons-ячейки* и составление *диаграммы представления списка в виде cons-ячеек*. Помимо определения новой функции при помощи макроса `defun`, так же были рассмотрены такие функции как:

- `quote` или `'` — блокирует вычисление выражения;
- `car` — возвращает *car-часть cons-ячейки*;
- `cdr` — возвращает *cdr-часть cons-ячейки*;
- `cons` — создаёт новую *cons-ячейку*, у которой *car-часть* принимает первый аргумент, а *cdr-часть* — второй аргумент;
- `list` — возвращает *список* из элементов переданных ей в качестве аргументов.