



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»

## **Лабораторная работа № 6**

Студент Лучина Е.Д

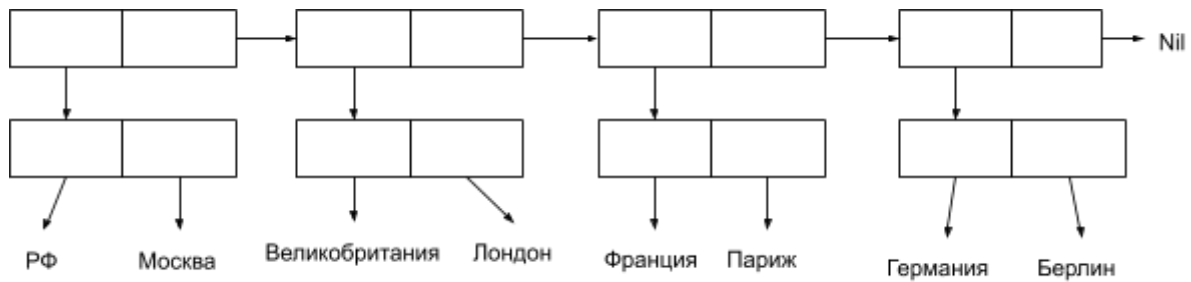
Группа ИУ7-61Б

Преподаватель Толпинская Н.Б.

Москва.  
2020 г.

Структура - набор элементов типа Страна-Столица. Найти столицу по стране  
Найти страну по столице.

### 1) Представление в виде списка точечных пар



Для M стран занимает  $2 * M * \text{sizeof}(\text{списковая ячейка})$  памяти.

```
(setf lofpairs (list (cons 'Russia 'Moscow) (cons 'France 'Paris) (cons 'Germany 'Berlin) (cons 'UK 'London)))
```

```
((RUSSIA . MOSCOW) (FRANCE . PARIS) (GERMANY . BERLIN) (UK . LONDON))
```

```
(defun p_findcity (list country)
  (if
    list
    (if (eq (caar list) country) (cdar list) (p_findcity (cdr list) country))
  )
)
```

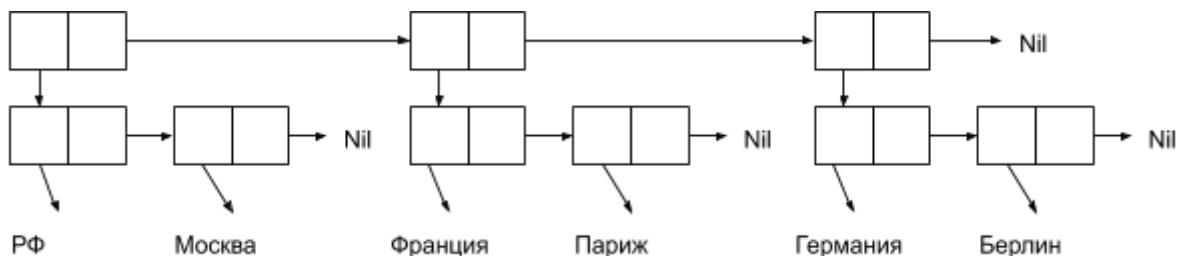
```
(defun p_findcountry (list city)
  (if
    list
    (if (eq (cdar list) city) (caar list) (p_findcountry (cdr list) city))
  )
)
```

```
(print (p_findcity lofpairs 'France)) -> PARIS
```

```
(print (p_findcountry lofpairs 'Berlin)) -> GERMANY
```

cpu time: 0.07 sec

### 2) Представление в виде списка двухэлементных списков



Для М стран занимает  $3 * M * \text{sizeof(списковая ячейка)}$  памяти.

```
(setf loflists (list (list 'Russia 'Moscow) (list 'France 'Paris) (list 'Germany 'Berlin) (list 'UK  
'London)))
```

```
((RUSSIA MOSCOW) (FRANCE PARIS) (GERMANY BERLIN) (UK LONDON))
```

```
(defun l_findcity (list country)  
  (if  
    list  
    (if (eq (caar list) country) (cadar list) (l_findcity (cdr list) country))  
    )  
  )
```

```
(defun l_findcountry (list city)  
  (if  
    list  
    (if (eq (cadar list) city) (caar list) (l_findcountry (cdr list) city))  
    )  
  )
```

```
(print (l_findcity loflists 'France)) -> PARIS  
(print (l_findcountry loflists 'Berlin)) -> GERMANY  
cpu time: 0.07 sec
```

Первый вариант представления занимает меньше памяти и процессорного времени, за счет меньшего количества переходов по указателям. Момент сравнения стран одинаков, получение столицы требует в первом случае один переход по указателям, во втором - два. Поиск по столице существенно сложнее в списке списков, чем в точечной паре. Каждое сравнение требует перехода по трем указателям, когда в варианте точечных пар - по двум. Получение названия страны одинаково. Так что если точно известно, что в качестве элемента списка используется пара значений, выгоднее использовать список точечных пар.