

ГУАП

КАФЕДРА № 43

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ  
ПРЕПОДАВАТЕЛЬ

кандидат технических наук  
\_\_\_\_\_  
должность, уч. степень, звание

\_\_\_\_\_  
подпись, дата

П.А. Степанов  
\_\_\_\_\_  
инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №5  
**Решение логических задач**

по дисциплине: Функциональное и логическое программирование

РАБОТУ ВЫПОЛНИЛ  
СТУДЕНТ ГР.

4134к  
\_\_\_\_\_

\_\_\_\_\_  
подпись, дата

Костяков Н.А.  
\_\_\_\_\_  
инициалы, фамилия

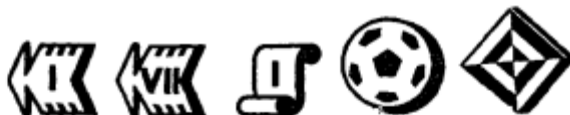
Санкт-Петербург  
2023

## Цель работы

Составить программу для решения одной из логических задач № 1–4, 19, 20, 27, 31, 33, 40–43, 56, 58, 59, 61, 62, 64–67, 76–85, условия которых приведены в книге: Бизам Д., Герцер Я. Игра и логика. — М.: Мир, 1975 г.

## Задача 4

### 4. БЛЕСТЯЩИЕ ОФИЦЕРЫ



На одном вечере среди гостей оказалось пять офицеров: пехотинец, артиллерист, летчик, связист и сапер. Один из них был капитаном, трое — майорами и один — в звании подполковника. Дамы окружили офицеров таким вниманием, что все остальные гости оказались просто забытыми. Из разговоров удалось выяснить следующее:

- 1) у Яноша такое же звание, как и у его друга сапера;
- 2) офицер-связист и Ференц — большие друзья;
- 3) офицер-летчик вместе с Белой и Лайошем недавно побывали в гостях у Ференца;
- 4) незадолго до званого вечера у артиллериста и сапера почти одновременно вышли из строя радиопри-

19

емники. Оба в один день обратились к Лайошу с просьбой зайти к ним и помочь связисту устранить неисправность и не ошиблись, поскольку с тех пор приемники у обоих работают отлично;

5) Ференц чуть было не стал летчиком, но потом по совету своего друга сапера избрал иной род войск;

6) Янош по званию старше Лайоша, а Бела старше Ференца;

7) пятый офицер, Андраш, накануне вечера был в гостях у Лайоша.

Определите звание каждого офицера и род войск, в котором он служит.

(Задача взята из журнала *A Középiskolai Matematikai Lapok* — Математика в средней школе.)



## Листинг программы

```
same_rank(yanosh, saper).
same_rank(saper, yanosh).

friend(saper, yanosh).
friend(yanosh, saper).
```

```

friend(linkman, ferenc).
friend(ferenc, linkman).
friend(saper, ferenc).
friend(ferenc, saper).
friend(andrash, layosh).

not_pilot(ferenc).
not_pilot(bela).
not_pilot(layosh).

not_linkman(ferenc).
not_linkman(layosh).

not_artilerian(layosh).

not_saper(layosh).
not_saper(ferenc).

grater(yanosh, layosh).
grater(bela, ferenc).

can_be_colonel(X):- not(same_rank(X, Y)).
lower_rank(X):- not(grater(X, Y)).

colonel(X):- can_be_colonel(X), not(lower_rank(X)).

major(X):- (same_rank(X, Y)); grater(Z, X), colonel(Z).

captain(X):- not(same_rank(X, Y)), grater(Z,X), major(Z).

pilot(X):- not(not_pilot(X)), not(marine(X)).

saper(X):- not(not_saper(X)), not(grater(X, Z)), not(pilot(X)), not(marine(X)),
not(artilerian(X)).

linkman(X):- not(saper(X)), not(pilot(X)), not(not_linkman(X)).

marine(X):- not_saper(X), not_artilerian(X), not_linkman(X), not_pilot(X).

artilerian(X):- not(saper(X)), not(pilot(X)), not(not_artilerian(X)),
not(marine(X)), not(linkman(X)).

check_name(Name, Predicate) :-
    current_predicate(Predicate/1), % проверяем, что предикат существует
    call(Predicate, Name), % вызываем предикат с именем в качестве аргумента
    write(Name), write(' is a '), write(Predicate), nl. % выводим результат в
консоль

```

```
% создание списка имен и предикатов
names([bela, ferenc, layosh, andrash, yanosh]).
predicates([colonel, major, captain, artillerian, saper, linkman, marine, pilot]).

% цикл по спискам
loop:-
    names(Names), % получаем список имен
    predicates(Predicates), % получаем список предикатов
    member(Name, Names), % выбираем имя из списка
    member(Predicate, Predicates), % выбираем предикат из списка
    check_name(Name, Predicate),
    fail.
```

## Вывод программы

```
1 ?- loop.
bela is a colonel
bela is a linkman
ferenc is a major
ferenc is a artillerian
layosh is a captain
layosh is a marine
andrash is a pilot
yanosh is a major
yanosh is a pilot
false.

2 ?- []
```