

Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Сибирский государственный университет телекоммуникаций и информатики»  
(СибГУТИ)

Кафедра прикладной математики и кибернетики

Лабораторная работа №5  
по дисциплине «Функциональное и логическое программирование»

Бригада №1

Выполнил:

студенты группы ИП-813  
Стояк Юрий Константинович  
Бурдуковский Илья Александрович

Работу проверил:

ассистент кафедры ПМиК  
Пащенко Анастасия Андреевна

Новосибирск 2020 г.

## Задание

1. Написать предикат, который печатает все нечётные числа из диапазона в порядке убывания. Границы диапазона вводятся с клавиатуры в процессе работы предиката.

2. Написать предикат, который находит числа Фибоначчи по их номерам, которые в цикле вводятся с клавиатуры. Запрос номера и нахождение соответствующего числа Фибоначчи должно осуществляться до тех пор, пока не будет введено отрицательное число. Циклический ввод организовать с помощью предиката repeat. Числа Фибоначчи определяются по следующим формулам:  $F(0)=1$ ,  $F(1)=1$ ,  $F(i)=F(i-2)+F(i-1)$  ( $i=2, 3, 4, \dots$ ).

3. Написать предикат, который разбивает числовой список по двум числам, вводимым с клавиатуры на три списка: меньше меньшего введенного числа, от меньшего введенного числа до большего введенного числа, больше большего введенного числа. Список и два числа вводятся с клавиатуры в процессе работы предиката. Например:  $[3,7,1,-3,5,8,0,9,2]$ , 8, 3  $\rightarrow [1,-3,0,2]$ ,  $[3,7,5,8]$ ,  $[9]$ .

4. Написать предикат, который формирует список из наиболее часто встречающихся элементов списка. Список вводится с клавиатуры в процессе работы предиката. Встроенные предикаты поиска максимума и сортировки не использовать! Например:  $[0,3,5,7,1,5,3,0,3,3,5,7,0,5,0] \rightarrow [0,3,5]$ .

## Листинг

```
% Автор: Я
% Дата: 18.11.2020
% -----1-----

task1:-write('print X:'),read(X),write('print
Y:'),read(Y),task1(X,Y).
task1(X, Y):- (Y >= X -> ((Y mod 2) == 1) -> write(Y); write('
')),
Z is (Y - 1),task1(X, Z); true).

% -----2-----

task2:-repeat,writeln('Enter number'),read(N),fib(N),!.
f(0,1):-!.
f(1,1):-!.
f(I,F):-I1 is I-1,I2 is I-2,f(I1,F1),f(I2,F2),F is F1+F2.
fib(N):-N<0,!.
fib(N):-f(N, F),writeln(F),fail.

% -----3-----

task3:-writeln('Enter numbers'),read(A),read(B),
        writeln('Enter list'),read(L),

compare_AB(A,B,L,L1,L2,L3),writeln(L1),writeln(L2),writeln(L3).
compare_AB(A,B,L,L1,L2,L3):- (A<B ->split_1(A,B,L,L1,L2,L3);
split_1(B,A,L,L1,L2,L3)).
split_1(A,B,[H|T],[H|T1],L2,L3):- (H<A -
>split_1(A,B,T,T1,L2,L3)).
split_1(A,B,[H|T],L1,L2,[H|T3]):- (H>B -
>split_1(A,B,T,L1,L2,T3)).
split_1(A,B,[H|T],L1,[H|T2],L3):-split_1(A,B,T,L1,T2,L3).
split_1(_,_,[],[],[],[]).

% -----4-----

move_max_to_end([], []):-!.
```

```

move_max_to_end([Head], [Head]):-!.
move_max_to_end([First, Second|Tail],
    [Second|ListWithMaxEnd]):-First > Second, !,
    move_max_to_end([First|Tail], ListWithMaxEnd).
move_max_to_end([First, Second|Tail],
    [First|ListWithMaxEnd]):- move_max_to_end([Second|Tail],
ListWithMaxEnd).

sorting(SortedList, SortedList):- move_max_to_end(SortedList,
DoubleSortedList),
    SortedList = DoubleSortedList, !.
sorting(List, SortedList):- move_max_to_end(List, SortedPart),
    sorting(SortedPart, SortedList).

task4:-write("Список:"), nl, read(ListA),
    sorting(ListA, Y),task4(Y, Z), write(Z).

task4([], []).
task4([H|T], Res):-task4([H|T], _, Res, _).

task4([], 0, [], 0).
task4([H], 1, [H], 1).
task4([H, H1|T], Count, Res, Max):-task4([H1|T], Count1, Res1,
Max1),
    Count2 is Count1 + 1,
    ((H == H1)->
        (Count = Count2, ((Count2 > Max1)->
            Max = Count2, Res = [H];
            (Max = Max1, ((Count2 == Max1)->append([H], Res1,
Res);Res = Res1)))
        );
    Count = 1,Max = Max1,Res = Res1).

```

## Результаты работы

```
print X:-10.  
print Y:|: 10.  
 9 7 5 3 1 -1 -3 -5 -7 -9  
true.  
  
?- task2.  
Enter number  
|: 7.  
21  
Enter number  
|: -1.  
  
true.  
  
?- task3.  
Enter numbers  
|: 3.  
|: 8.  
Enter list  
|: [3,7,1,-3,5,8,0,9,2].  
[1,-3,0,2]  
[3,7,5,8]  
[9]  
true.  
  
?- task4.  
Список:  
|: [0,3,5,7,1,5,3,0,3,3,5,7,0,5,0].  
[0,3,5]  
true.  
.
```