



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»

НАПРАВЛЕНИЕ ПОДГОТОВКИ \_\_\_\_\_ «09.03.04 Программная инженерия»

## ОТЧЕТ

### ПО ЛАБОРАТОРНЫМ РАБОТАМ №16-18

Дисциплина: \_\_\_\_\_ Функциональное и логическое программирование

Студент	<u>ИУ7-66Б</u>	_____	<u>Т. А. Казаева</u>
	Группа	Подпись, дата	И. О. Фамилия

Преподаватель	_____	<u>Н. Б. Толпинская</u>
	Подпись, дата	И. О. Фамилия

Москва, 2022 г.

# ЛАБОРАТОРНАЯ РАБОТА 16

Используя хвостовую рекурсию, разработать программу, позволяющую найти:

а)  $n!$ ,

б)  $n$ -е число Фибоначчи.

```
1 predicates
2   Factorial(integer, integer).
3   FactorialHelper(integer, integer, integer).
4
5   Fib(integer, integer).
6   FibHelper(integer, integer, integer, integer).
7
8 clauses
9   FactorialHelper(N, Res, Acc) :- N <= 1, Res = Acc, !.
10  FactorialHelper(N, Res, Acc) :- UpdAcc = Acc * N, UpdN = N - 1,
    FactorialHelper(UpdN, Res, UpdAcc).
11  Factorial(N, Res) :- FactorialHelper(N, Res, 1).
12
13  FibHelper(N, Res, _, Acc) :- N <= 2, Res = Acc, !.
14  FibHelper(N, Res, Prev, Acc) :- UpdN = N - 1, UpdAcc = Prev + Acc, FibHelper
    (UpdN, Res, Acc, UpdAcc).
15  Fib(N, Res) :- FibHelper(N, Res, 1, 1).
16
17 goal
18   %Factorial(3, Ans).
19   Fib(3, Ans).
```

# шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: T1 = T2 и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Factorial(3, Ans), FactorialHelper(N, Res, Acc)	Откат, переход к следующему предложению
...			
3	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Factorial(3, Ans), Factorial(N, Res)	Прямой ход, новое состояние резольвенты
4	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(3, Ans, 1), FactorialHelper(N, Res, Acc)	Прямой ход, новое состояние резольвенты
5	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(3, Ans, 1), FactorialHelper(N, Res, Acc)	Откат к предыдущему состоянию резольвенты Новая подстановка: {N = 3, Res = Ans}
6	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(3, Ans, 1), FactorialHelper(N, Res, Acc)	Прямой ход, новое состояние резольвенты
7	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(3, Ans, 1), FactorialHelper(N, Res, Acc)	Прямой ход, новое состояние резольвенты
8	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(3, Ans, 1), FactorialHelper(N, Res, Acc)	Прямой ход, новое состояние резольвенты
9	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(3, Ans, 1), FactorialHelper(N, Res, Acc)	Прямой ход, новое состояние резольвенты
10	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(3, Ans, 1), FactorialHelper(N, Res, Acc)	Откат к предыдущему состоянию резольвенты Новая подстановка: {N = 2, Res = Ans, Acc = 3}
11	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(2, Ans, 3), FactorialHelper(N, Res, Acc)	Прямой ход, новое состояние резольвенты
12	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(2, Ans, 3), FactorialHelper(N, Res, Acc)	Прямой ход, новое состояние резольвенты
13	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(2, Ans, 3), FactorialHelper(N, Res, Acc)	Прямой ход, новое состояние резольвенты

# шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: T1=T2 и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	$1 \leq 1$ Res = 6		
15	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека Res = 6	1 <= 1 верно	Прямой ход, новое состояние резольвенты
16	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Унификация: Ans = 6	Прямой ход, новое состояние резольвенты
17	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Успешно, подстановка: {Ans = 6}	Резольвента пуста. Вывод: Ans = 6 Откат с отсечением остаточных предложений процедуры относительно шага 14 Новая подстановка: {N = 2, Res = Ans, Acc = 3}
18	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(1, Ans, 6), Factorial(N, Res)	Откат, переход к следующему предложению
...		Унификация неуспешна(несовпадение функторов)	
21	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(1, Ans, 6), Fib(N, Res)	Конец БЗ достигнут. Откат, переход к следующему предложению относительно пункта 11 Новая подстановка: {N = 2, Res = Ans, Acc = 1}
22	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна(несовпадение функторов)  FactorialHelper(2, Ans, 3), Factorial(N, Acc) Унификация неуспешна(несовпадение функторов)	Откат, переход к следующему предложению
...			
25	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(2, Ans, 3), Fib(N, Res)	Конец БЗ достигнут. Откат, переход к следующему предложению относительно пункта 11 Новая подстановка: {N = 3, Res = Ans, Acc = 1}
26	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна(несовпадение функторов)  FactorialHelper(3, Ans, 1), Factorial(N, Res)	Откат, переход к следующему предложению
...		Унификация неуспешна (несовпадение функторов)	
29	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	FactorialHelper(3, Ans, 1), Fib(N, Res)	Конец БЗ достигнут. Откат, переход к следующему предложению относительно пункта 3
30	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна(несовпадение функторов)  Factorial(3, Ans) FibHelper(N, Res, _', Acc)	Откат, переход к следующему предложению
...		Унификация неуспешна(несовпадение функторов)	
32	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Factorial(3, Ans) Fib(N, Res)	Конец БЗ достигнут. Вывод результата

# шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: T1=T2 и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Fib(3, Ans). FactorialHelper(N, Res, Acc)	Откат, переход к следующему предложению
...			
6	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Fib(3, Ans). Fib(N, Res)	Прямой ход, новое состояние резольвенты
7	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация успешна: подстановка: {N = 3, Res = Ans} FibHelper(3, Res, 1, 1). FactorialHelper(N, Res, Acc)	Откат, переход к следующему предложению
...			
10	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	FibHelper(3, Res, 1, 1). FibHelper(N, Res, _, Acc)	Прямой ход, новое состояние резольвенты
11	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация успешна: подстановка {N = 3, Res = Ans, Acc = 1}  3 <= 2, Res = 1, !	Откат к предыдущему состоянию резольвенты Новая подстановка: {N = 3, Res = Ans}
12	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	FibHelper(3, Res, 1, 1). FibHelper(N, Res, Prev, Acc)	Прямой ход, новое состояние резольвенты
13	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация успешна: подстановка {N = 3, Res = Ans, Prev = 1, Acc = 1}  Унификация: UpdN = 3 - 1  Успешно, подстановка: {UpdN = 2}	Новое состояние резольвенты
14	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация: UpdAcc = 1 + 1 FibHelper(UpdN, Ans, 1, UpdAcc)	Новое состояние резольвенты
15	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Успешно, подстановка: {UpdAcc = 2} FibHelper(2, Ans, 1, 2) FactorialHelper(N, Res, Acc)	Откат, переход к следующему предложению
...			
18	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	FibHelper(2, Ans, 1, 2) FibHelper(N, Res, _, Acc)	Новое состояние резольвенты
19	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация успешна, подстановка {N = 2, Res = Ans, Acc = 2}  2 <= 2 Res = Acc !	Новое состояние резольвенты

# шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: $T_1 = T_2$ и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
	$Res = 2$	Унификация: $Res = 2$	
20	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Успешно, подстановка: $\{Res = 2\}$	Новое состояние резольвенты
	!		Резольвента пуста.
21	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Встречен системный предикат отсечения	Вывод: $Ans = 2$ Откат с отсечением остаточных предложений процедуры относительно шага 18
	!		Новая подстановка: $\{N = 2, Res = Ans\}$
22	$FibHelper(2, Ans, 1, 2)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$FibHelper(2, Ans, 1, 2)$ $Fib(N, Res)$	Конец БЗ достигнут. Откат, переход к следующему предложению относительно пункта 12
	$FibHelper(3, Ans, 1, 1)$	Унификация неуспешна (несоответствие факторов) $FibHelper(3, Ans, 1, 1)$ $Fib(N, Res)$	Новая подстановка: $\{N = 3, Res = Ans\}$
23	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несоответствие факторов)	Конец БЗ достигнут. Откат, переход к следующему предложению относительно пункта 6
	$Fib(3, Ans)$	-	Конец БЗ достигнут. Вывод результата на экран
24	Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	-	

# ЛАБОРАТОРНАЯ РАБОТА 17

Используя хвостовую рекурсию, разработать эффективную программу (комментируя назначение аргументов), позволяющую:

- а) Найти длину списка (по верхнему уровню);
- б) Найти сумму элементов числового списка;
- с) Найти сумму элементов числового списка, стоящих на нечетных позициях исходного списка (нумерация от 0);

```
1 domains
2   list = integer *.
3 predicates
4   ListLen(list , integer) .
5   ListLenHelper(list , integer , integer) .
6
7   ListSum(list , integer) .
8   ListSumHelper(list , integer , integer) .
9
10  ListOddSum(list , integer) .
11  ListOddSumHelper(list , integer , integer) .
12
13 clauses
14  ListLenHelper ([] , Res , Acc) :- Res = Acc , !.
15  ListLenHelper ([_|T] , Res , Acc) :- UpdAcc = Acc + 1 , ListlenHelper(T , Res ,
    UpdAcc) .
16  ListLen(List , Res) :- ListLenHelper(List , Res , 0) .
17
18  ListSumHelper ([] , Res , Acc) :- Res = Acc , !.
19  ListSumHelper ([H|T] , Res , Acc) :- UpdAcc = Acc + H , ListSumHelper(T , Res ,
    UpdAcc) .
20  ListSum(List , Res) :- ListSumHelper(List , Res , 0) .
21
22  ListOddSumHelper ([] , Res , Acc) :- Res = Acc , !.
23  ListOddSumHelper ([H|[_|T]] , Res , Acc) :- UpdAcc = Acc + H , ListOddSumHelper(
    T , Res , UpdAcc) , !.
24  ListOddSumHelper ([H|_] , Res , Acc) :- Res = Acc + H.
25
26  ListOddSum(List , Res) :- ListOddSumHelper(List , Res , 0) .
27
28 goal
29  ListSum([3 , 4 , 5] , Res) .
```

# шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: T1—T2 и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSum([3, 4, 5], Res) Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
...			
6	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSum([3, 4, 5], Res) Унификация успешна: подстановка {List = [3, 4, 5], Res = Res, Acc = 0}	Новое состояние резольвенты
7	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSumHelper([3, 4, 5], Res, 0), ListLenHelper([], Res, Acc) Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
...			
10	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSumHelper([3, 4, 5], Res, 0), ListSumHelper([], Res, Acc) Унификация неуспешна (несовпадение первых аргументов)	Откат, переход к следующему предложению
11	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSumHelper([3, 4, 5], Res, 0), ListSumHelper([H T], Res, Acc) Унификация успешна, подстановка {H = 1, List = [4, 5], Res = Res, Acc = 0}	Новое состояние резольвенты, прямой ход
12	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	UpdAcc = 0 + 3 ListSumHelper([4, 5], Res, UpdAcc) Успешно, подстановка: {UpdAcc = 3}	Новое состояние резольвенты, прямой ход
13	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSumHelper([4, 5], Res, UpdAcc) Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
...			
16	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSumHelper([4, 5], Res, UpdAcc) Унификация успешна, подстановка {H = 4, List = [5], Res = Res, Acc = 3}	Новое состояние резольвенты, прямой ход
17	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	UpdAcc = 3 + 4 ListSumHelper([5], Res, UpdAcc) Успешно, подстановка: {UpdAcc = 7}	Новое состояние резольвенты, прямой ход
18	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSumHelper([5], Res, 7), ListLenHelper([], Res, Acc) Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
...			
22	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSumHelper([5], Res, 7) Унификация успешна, подстановка {H = 5, List = [], Res = Res, Acc = 7}	Новое состояние резольвенты, прямой ход
23	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	UpdAcc = 7 + 5 ListSumHelper([], Res, UpdAcc) Успешно, подстановка: {UpdAcc = 12}	Новое состояние резольвенты, прямой ход
24	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	ListSumHelper([], Res, 12), ListLenHelper([], Res, Acc) Унификация неуспешна (несовпадение функторов)	Откат, переход к следующему предложению
...			



# шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: $T_1 = T_2$ и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
	<code>ListSumHelper([], Res, 12)</code>	<code>ListSumHelper([], Res, UpdAcc), ListSumHelper([], Res, Acc)</code>	Новое состояние резольвенты, прямой ход
27	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация успешна, подстановка $\{Res = Res, Acc = 12\}$	Новое состояние резольвенты, прямой ход
	!	Унификация: $Res = 12$	
28	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Успешно, подстановка: $\{Res = 12\}$	Новое состояние резольвенты, прямой ход
	!		Резольвента пуста.
29	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Встречен системный предикат отсечения	Запомнить $Res = 12$
			Откат с отсечением остальных предложений процедуры относительно шага 27
	<code>ListSumHelper([], Res, 12)</code>	<code>ListSumHelper([], Res, 12), ListSumHelper([H T], Res, Acc)</code>	Откат, переход к следующему предложению
30	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несовпадение первых аргументов)	
...			
	<code>ListSumHelper([], Res, 12)</code>	<code>ListSumHelper([], Res, 12), ListOddSum(List, Res)</code>	Конец БЗ достигнут. Откат, переход к следующему предложению относительно пункта 22
35	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несовпадение функций)	Новая подстановка: $\{List = [5], Res = Res, Acc = 7\}$
	<code>ListSumHelper([5], Res, 7)</code>	<code>ListSumHelper([5], Res, 7), ListSum(List, Res)</code>	Откат, переход к следующему предложению
36	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несовпадение функций)	
...			
	<code>ListSumHelper([5], Res, 7)</code>	<code>ListSumHelper([5], Res, 7), ListOddSum(List, Res)</code>	Конец БЗ достигнут. Откат, переход к следующему предложению относительно пункта 16
40	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несовпадение функций)	Новая подстановка: $\{List = [4, 5], Res = Res, Acc = 3\}$
	<code>ListSumHelper([4, 5], Res, 3)</code>	<code>ListSumHelper([4, 5], Res, 3), ListSum(List, Res)</code>	Откат, переход к следующему предложению
41	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несовпадение функций)	
...			
	<code>ListSumHelper([4, 5], Res, 3)</code>	<code>ListSumHelper([4, 5], Res, 3), ListOddSum(List, Res)</code>	Конец БЗ достигнут. Откат, переход к следующему предложению относительно пункта 6
45	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несовпадение функций)	Новая подстановка: $\{List = [3, 4, 5], Res = Res, Acc = 0\}$
	<code>ListSumHelper([3, 4, 5], Res, 0)</code>	<code>ListSumHelper([3, 4, 5], Res, 0), ListSum(List, Res)</code>	Откат, переход к следующему предложению
46	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несовпадение функций)	
...			
	<code>ListSumHelper([3, 4, 5], Res, 0)</code>	<code>ListSumHelper([3, 4, 5], Res, 0), ListOddSum(List, Res)</code>	Конец БЗ достигнут. Вывод результата на экран
50	Резольвента не пустая, попытка унификации для подцели, извлекаемой из стека	Унификация неуспешна (несовпадение функций)	

# ЛАБОРАТОРНАЯ РАБОТА 18

Используя хвостовую рекурсию, разработать, комментируя аргументы, эффективную программу, позволяющую:

- a) Сформировать список из элементов числового списка, больших заданного значения;
- b) Сформировать список из элементов, стоящих на нечетных позициях исходного списка (нумерация от 0);
- c) Удалить заданный элемент из списка (один или все вхождения);
- d) Преобразовать список в множество (можно использовать ранее разработанные процедуры).

```
1 domains
2   list = integer*.
3 predicates
4   Append(list , list , list).
5   ListGreaterThen(list , integer , list).
6   ListLeaveOdd(list , list).
7   ListRemoveElem(list , integer , list).
8   ListIncludes(list , integer).
9   ListToSet(list , list).
10 clauses
11   Append([_|T] , List , [_|Res]) :- Append(T , List , Res) , !.
12   Append([], List , List).
13   ListGreaterThen([H|T] , N , [H|Res]) :- H > N , ! , ListGreaterThen(T , N , Res) ,
14     !.
15   ListGreaterThen([_|T] , N , Res) :- ListGreaterThen(T , N , Res) , !.
16   ListGreaterThen([], _ , []).
17   ListLeaveOdd([H|_|T]) , [H|Res]) :- ListLeaveOdd(T , Res) , !.
18   ListLeaveOdd(List , List).
19   ListRemoveElem([H|T] , Item , Res) :- Item = H , ! , ListRemoveElem(T , Item , Res
20     ).
21   ListRemoveElem([H|T] , Item , [H|Res]) :- ListRemoveElem(T , Item , Res) , !.
22   ListRemoveElem(List , _ , List).
23   ListIncludes([_|T] , Item) :- ListIncludes(T , Item) , !.
24   ListIncludes([H|_] , H).
25   ListToSet([H|T] , Res) :- ListIncludes(T , H) , ! , ListToSet(T , Res).
26   ListToSet([H|T] , [H|Res]) :- ListToSet(T , Res) , !.
27   ListToSet(List , List).
28 goal
29   ListGreaterThen([3 , 4 , 5] , 4 , List).
```

# шага	Состояние резольвенты, и вывод: дальнейшие действия (почему?)	Для каких термов запускается алгоритм унификации: $T1 = T2$ и каков результат (и подстановка)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
1	$ListGreaterThen([3, 4, 5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([3, 4, 5], 4, List), Append([\_T], List, \_Res)$ Унификация успешна (несоименитие функций)	Откат, переход к следующему предложению
...			
3	$ListGreaterThen([3, 4, 5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([3, 4, 5], 4, List), ListGreaterThen([H[T], N, [H[Res]])$ Унификация успешна, подстановка $\{H = [3], T = [4, 5], N = 4, [H[Res] = List\}$	Новое состояние резольвенты, прямой ход
4	$ListGreaterThen([4, 5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	1 < 4 неверно	Откат, возврат к предыдущему состоянию резольвенты, переход к следующему предложению
5	$ListGreaterThen([4, 5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([3, 4, 5], 4, List), ListGreaterThen([\_T], N, Res)$ Унификация успешна, подстановка $\{T = [4, 5], N = 4, Res = List\}$	Новое состояние резольвенты, прямой ход
6	$ListGreaterThen([4, 5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([4, 5], 4, List), Append([\_T], List, \_Res)$ Унификация успешна (несоименитие функций)	Откат, переход к следующему предложению
...			
8	$ListGreaterThen([4, 5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([3, 4, 5], 4, List), ListGreaterThen([H[T], N, Res)$ Унификация успешна, подстановка $\{H = 4, T = [5], N = 4, Res = List\}$	Новое состояние резольвенты, прямой ход
9	$ListGreaterThen([5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	4 < 4 неверно	Откат, возврат к предыдущему состоянию резольвенты, переход к следующему предложению
10	$ListGreaterThen([4, 5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([4, 5], 4, List), ListGreaterThen([\_T], N, Res)$ Унификация успешна, подстановка $\{T = [5], N = 4, [4Res] = List\}$	Новое состояние резольвенты, прямой ход
11	$ListGreaterThen([5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([5], 4, List), Append([\_T], List, \_Res)$ Унификация успешна (несоименитие функций)	Откат, переход к следующему предложению
...			
13	$ListGreaterThen([5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([5], 4, List), ListGreaterThen([H[T], N, Res)$ Унификация успешна, подстановка $\{H = 5, T = [], N = 4, [5]Res = List\}$	Новое состояние резольвенты, прямой ход
14	$ListGreaterThen([], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	5 < 4 верно	Новое состояние резольвенты, прямой ход
15	$ListGreaterThen([], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	Встречен системный предикат отсечения	Дальнейшие предложения процедуры рассматриваться не будут. Новое состояние резольвенты, прямой ход
16	$ListGreaterThen([], 4, List)$ Резольвента не пуста, попытка унификации для подцели, извлекаемой из стека	$ListGreaterThen([], 4, List), Append([\_T], List, \_Res)$ Унификация успешна (несоименитие функций)	Откат, переход к следующему предложению
...			

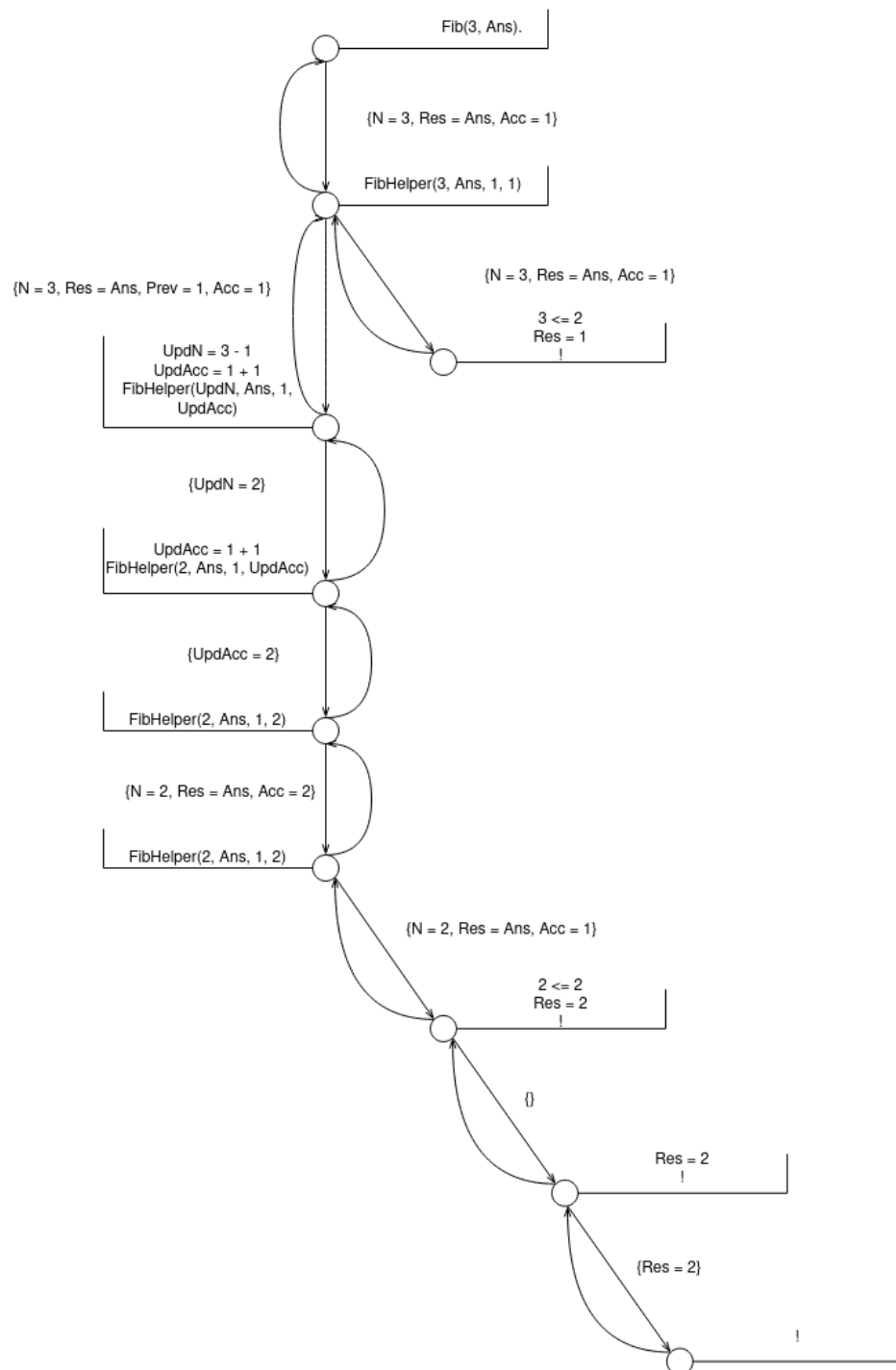
# шаг	Состояние резольвенты, и вывод дальнейшего действия (почему?)	Дальнейшие действия: прямой ход или откат (к чему приводит?)
	$ListGreaterThen([], 4, List)$	Резольвента пуста.
20	Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Вывод результата на экран: [5]
	$ListGreaterThen([], 4, List)$	Откат, переход к следующему предложению
21	Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Откат, переход к следующему предложению
...		
33	$ListGreaterThen([], 4, List)$ Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Конец B3 достигнут. Откат, переход к следующему предложению относительно пункта 13
	$ListGreaterThen([5], 4, List)$	
34	Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Откат, переход к следующему предложению
...		
46	$ListGreaterThen([5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Конец B3 достигнут. Новое состояние резольвенты
47	Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Дальнейшие предложения процедуры рассматриваться не будут. Резольвента пуста, восстановление резольвенты относительно шага 10
	$ListGreaterThen([4, 5], 4, List)$	
48	Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Откат, переход к следующему предложению
...		
60	$ListGreaterThen([4, 5], 4, List)$ Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Конец B3 достигнут. Новое состояние резольвенты
61	Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Дальнейшие предложения процедуры рассматриваться не будут. Резольвента пуста, восстановление резольвенты относительно шага 5
	$ListGreaterThen([4, 5], 4, List)$	
62	Резольвента не пуста, попытка унификации для подцели, являющейся из стека	Конец B3 достигнут. Резольвента пуста, завершение работы

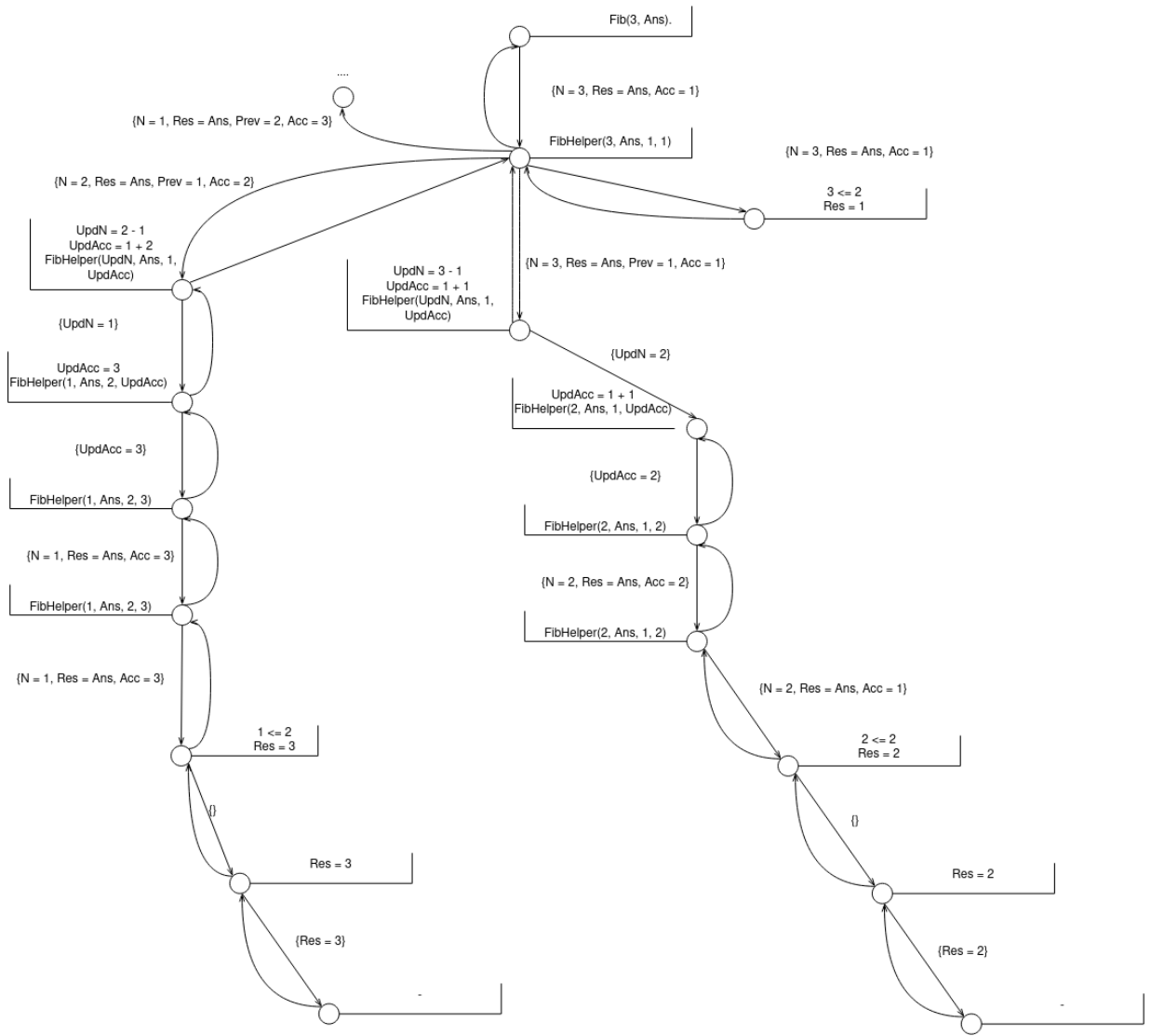
# ЗАЩИТЫ К ЛАБОРАТОРНЫМ РАБОТАМ

Построить дерево поиска решений для алгоритма нахождения n-го числа Фибоначчи:

- 1 С отсечением;
- 2 Без отсечения.

Как работает Prolog?(5 стеков).





Написать:

1 Свой reverse;

2 Программу для нахождения всех списков, которые можно получить из элементов заданного списка.

```

1 all_possible_lists(InList,Out) :- combinations(InList,_,SubList), permutations
  (SubList,Out).
2
3 combinations([], [], []).
4 combinations([H|T],[H|L],R) :- combinations(T,L,R).
5 combinations([H|T],L,[H|R]) :- combinations(T,L,R).
6
7 permutations([ ],[ ]) :- !.
8 permutations(L,[X|R]) :- omit(X,L,M), permutations(M,R).
9
10 omit(H,[H|T],T).
11 omit(X,[H|L],[H|R]) :- omit(X,L,R).

```

```

1 reverse ([],Z,Z) .
2 reverse ([H|T],Z,Acc) :- reverse (T,Z,[H|Acc]) .

```

Игорк находится в лабиринте MxN клеток. Лабиринт содержит стены и точки телепортации. Найти путь от точки старта до точки финиша.

```

1 w(0,0) .
2 w(0,1) . w(1,1) . w(2,1) . w(3,1) . w(4,1) . w(5,1) .
3 w(1,2) .          w(3,2) .          w(5,2) .
4 w(1,3) .          w(3,3) .          w(5,3) .
5 w(0,4) . w(1,4) . w(2,4) .          w(4,4) . w(5,4) .
6 w(2,5) . w(3,5) . w(4,5) .
7
8 t(w(1, 1) , w(3, 3)) .
9 t(w(3, 3) , w(5, 3)) .
10
11 path_exists(X0,Y0,X,Y) :- next_cell(X0,Y0,X,Y) , w(X,Y) .
12 next_cell(X0, Y0, X, Y) :- t(w(X0, Y0) , w(X_to, Y_to)) , X is X_to, Y is Y_to .
13 next_cell(X0,Y0,X0,Y) :- Y is Y0+1 .
14 next_cell(X0,Y0,X,Y0) :- X is X0+1 .
15 next_cell(X0,Y0,X0,Y) :- Y is Y0-1 .
16 next_cell(X0,Y0,X,Y0) :- X is X0-1 .
17
18 go(X,Y,X,Y,Path,Path) .
19 go(X0,Y0,X,Y,SoFar,Path) :-
20 path_exists(X0,Y0,X1,Y1) ,
21 \+ memberchk( w(X1,Y1) , SoFar) ,
22 go(X1,Y1,X,Y,[w(X1,Y1)|SoFar],Path) .

```