

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таныгин Максим Олегович
Должность: и.о. декана факультета фундаментальной и прикладной информатики
Дата подписания: 21.09.2023 12:56:21
Уникальный программный ключ:
65ab2aa0d384efe8480e6a4c688eddbc475e411a

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ

Проректор по учебной работе

О.Г. Локтионова

« 24 » 12



ОРГАНИЗАЦИЯ РЕКУРСИВНЫХ ВЫЧИСЛЕНИЙ НА ЯЗЫКЕ PROLOG

Методические указания по выполнению лабораторной работы
для студентов направления подготовки 09.03.04 «Программная
инженерия» по дисциплине «Функциональное и логическое
программирование»

Курск 2017

УДК 004.652

Составители: В.Г. Белов, Т.М. Белова

Рецензент

Кандидат технических наук, доцент кафедры программной инженерии ЮЗГУ И.Н. Ефремова

Организация рекурсивных вычислений на языке Prolog: методические указания по выполнению лабораторной работы по дисциплине "Функциональное и логическое программирование" для студентов направления подготовки 09.03.04 "Программная инженерия" / Юго-Зап. гос. ун-т; сост.: В.Г. Белов, Т.М. Белова, – Курск, 2017. – 32 с.: ил. 16, табл. 3, прил. 1.

Рассмотрены методы составления рекурсивных определений предикатов с использованием нисходящей и восходящей рекурсий в среде **VisualProlog**.

Материал предназначен для студентов направления подготовки 09.03.04 «Программная инженерия» специализации «Разработка программно-информационных систем»

Подписано в печать 24.12.17. Формат 60x84 1/16.

Усл. печ. л. 17. Уч.-изд. л. 16. Тираж 100 экз. Заказ 4380. Бесплатно.

Юго-Западный государственный университет

305040, Курск, ул. 50 лет Октября, 94.

СОДЕРЖАНИЕ

1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ.....	4
2. МЕТОДИКА СОСТАВЛЕНИЯ РЕКУРСИВНЫХ ОПРЕДЕЛЕНИЙ	4
2.1. Нисходящая рекурсия.....	4
2.2. Примеры составления определений с нисходящей рекурсией	4
2.3. Восходящая рекурсия	5
2.4. Примеры составления определений с восходящей рекурсией.....	6
2.5. Индивидуальные задания для выполнения лабораторной работы в среде Visual Prolog.....	8
3. СОДЕРЖАНИЕ ОТЧЕТА	18
4. КОНТРОЛЬНЫЕ ВОПРОСЫ К ЗАЩИТЕ ЛАБОРАТОРНОЙ РАБОТЫ..	18
Приложение.....	19

1. ЦЕЛЬ ЛАБОРАТОРНОЙ РАБОТЫ.

Целью лабораторной работы является приобретение навыков в составлении рекурсивных определений предикатов с использованием нисходящей и восходящей рекурсий.

2. МЕТОДИКА СОСТАВЛЕНИЯ РЕКУРСИВНЫХ ОПРЕДЕЛЕНИЙ

2.1. Нисходящая рекурсия

2.1.1. Определяются все объекты данных, входящие в задачу. Хотя бы одним объектом должна быть рекурсивная структура данных.

2.1.2. Для терминального (конечного, граничного) состояния рекурсивной структуры данных записывается предикат, который определяет значение объекта – результата при этом терминальном значении структуры. Если таких состояний несколько, то для каждого необходим свой предикат для терминального состояния. В большинстве случаев это просто факт, в котором результат равен нулю, а конечное состояние структуры – "пусто".

2.1.3. Записывается определение разрабатываемого предиката для рекурсивного вызова в следующей последовательности.

2.1.3.1. Для предотвращения заикливания записывается предикат определяющий терминальное значение структуры. Он может отсутствовать, если это условие автоматически проверяется при унификации с заголовком правила.

2.1.3.2. В структуре выделяется первый элемент и ее остаток. Это можно сделать или при помощи специального предиката или путем унификации с заголовком правила. Специальный предикат должен быть первым в конъюнкции целей.

2.1.3.3. Записывается рекурсивный вызов разрабатываемого предиката, у которого вместо исходной рекурсивной структуры записывается ее остаток, а вместо переменной результата будет записываться переменная, которая будет хранить результат решения задачи для остатка исходной рекурсивной структуры данных.

2.1.3.4. Записывается предикат, определяющий решение всей задачи в переменной результата на основании значения первого элемента исходной рекурсивной структуры данных и значения результата для остатка исходной рекурсивной структуры данных.

2.2. Примеры составления определений с нисходящей рекурсией

2.2.1. Общая методика

Начальный элемент структуры: `B`.

Начальный элемент остатка структуры: `B`.

Структура: `Tree`.

Остаток структуры: `Rez_Tree`.

Результат для всей структуры: `Rez`.

Результат остатка структуры: `Rez_Rest`.

Разрабатываемый предикат: $PN(Tree, Rez)$.

Терминальный случай: $PN(\text{пусто}, 0)$.

Значение переменной Rez в конечном случае: $Rez = 0$.

Рекурсивный вызов:

$PN(Tree, Rez) :-$

$DIV(Tree, B, Rest_Tree),$ /* Предикат, выделяющий B и $Rest_Tree$ из $Tree$. */

$PN(Rest_Tree, Rez_Rest),$ /* Предикат, определяющий решение в Rez_Rest для $Rest_Tree$. */

$SOL(B, Rez_Rest, Rez).$ /* Предикат, определяющий решение всей задачи в Rez для $Tree$. */

Схематически решение задачи при помощи нисходящей рекурсии показано на рис.2.1,2.2.

Рекурсивный вызов

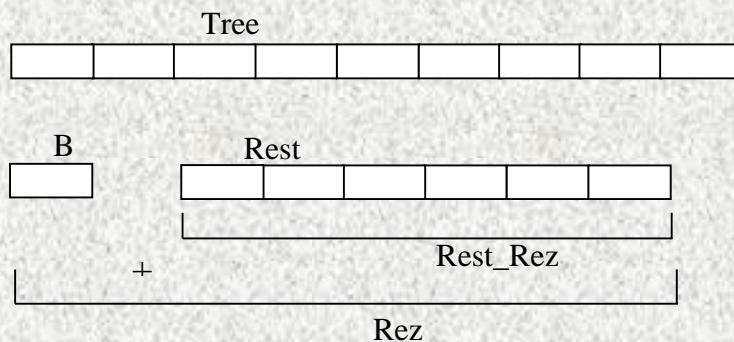


Рис. 2.1.

Терминальные случаи



Рис.2.2

2.2.2. Сумма в бинарном дереве (обратный обход дерева)

$SUM(g(B, L_Rest, R_Rest), Rez) :-$

$SUM(L_Rest, L_Rez),$

$SUM(R_Rest, R_Rez),$

$Res = B + L_Rez + R_Rez.$

Вызов:

$g(1, \text{пусто}, \text{пусто})$ или

$g(1, g(2, \text{пусто}, \text{пусто}), g(3, \text{пусто}, \text{пусто}))$

2.3. Восходящая рекурсия

- 2.3.1. Определяется список переменных вспомогательного предиката, в который помимо основных переменных разрабатываемого предиката вводятся дополнительные переменные, в одной из которых формируется результат разрабатываемого предиката.
- 2.3.2. Составляется определение вспомогательного предиката для терминального случая (или случаев), в котором значение результата будет равно значению дополнительной переменной, в которой формируется результат.
- 2.3.3. Для нетерминального случая в определении вспомогательного предиката записывается его вызов, в котором значения дополнительных переменных определяется на основании их предыдущих значений и значения очередного элемента в рекурсивной структуре данных.
- 2.3.4. Составляется определение разрабатываемого предиката, в которой осуществляется вызов вспомогательного предиката с начальными значениями вспомогательных переменных.

2.4. Примеры составления определений с восходящей рекурсией.

2.4.1. Общая методика

Структура данных: `Tree`.

Остаток структуры: `Rest_Tree`.

Начальный элемент структуры: `B`.

Начальный элемент остатка структуры: `B`.

Результат для всей структуры: `Rez`.

Результат от начала структуры: `Head_Rez`.

Результат для текущего рекурсивного вызова.

Разрабатываемый предикат: `PW (Tree, Rez)`.

Вспомогательный предикат: `PWT (Tree, Rez, Head_Rez)`.

Терминальный случай: `PWT (пусто, Rez, Rez)`.

При терминальном случае структура `Rez = Head_Rez_B`.

Вызов разрабатываемого вспомогательного предиката с начальным значением для `Head_Rez`:

```
PW (Tree, Rez) :-
```

```
PWT (Tree, Rez, 0).
```

Рекурсивный вызов вспомогательного предиката:

```
PWT (Tree, Rez, Head_Rez) :- /* Предикат, определяющий Rez, для
Tree на основании Head_Rez. */
```

```
  DIV (Tree, B, Rez_Tree), /* Предикат, выделяющий B и Rez_Tree из
Tree. */
```

```
  SOL (B, Head_Rez, Head_Rez_B), /* Предикат, определяющий
Head_Rez_B из Head_Rez и B. */
```

```
  PWT (Rest_Tree, Rez, Head_Rez_B). /* Предикат, определяющий
Rez, для Tree на основании Head_Rez_B, используя Rest_Tree.*/
```

Схематически процесс решения задачи при помощи восходящей рекурсии показан на рис.2.3 – 2.5.

Начальное состояние

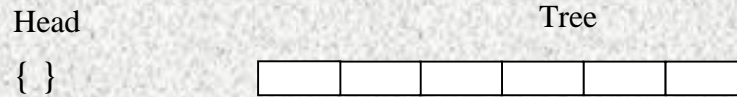


Рис. 2.3.

Рекурсивный вызов

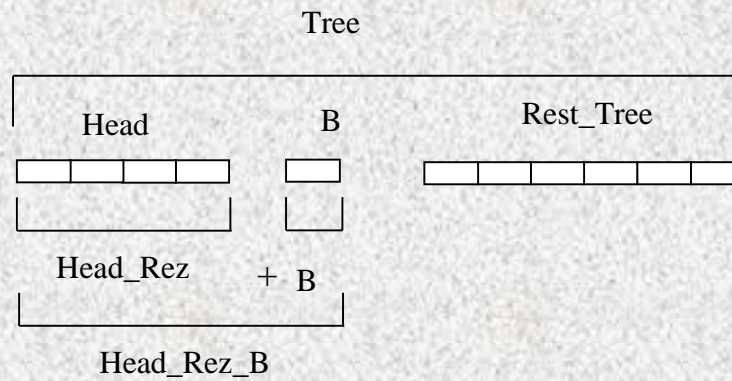


Рис. 2.4.

Последний шаг

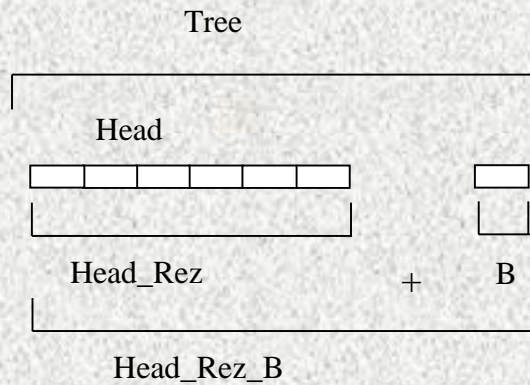


Рис. 2.5.

Tree

Следует учесть, что при применении восходящей рекурсии в алго

1. Пог списку из 10 разних типова малог инвентара. Погледајте ове мале инвентаре и одлучите се за један од њих. Овај инвентар је најбољи за вас.

3. Дан список из 15 целых чисел, среди которых могут быть одинаковые числа. Минимальные элементы этого списка замените целой частью среднего арифметического всех элементов списка, остальные члены списка оставить без изменения.

4. Дан список из 10 различных целых чисел. Найдите произведение чисел в этом списке, расположенных между максимальным и минимальным числами (в произведение включить оба эти числа).

5. Дан целочисленный список $a[15]$. Найдите наименьшее и наибольшее из четных чисел в последовательности чисел $a[15]$. Предусмотрите ситуацию, когда четных чисел в последовательности нет.

6. Дан целочисленный список $a[15]$. Найдите наибольшее из нечетных чисел и количество четных чисел в списке $a[15]$. Предусмотрите ситуацию, когда нечетных чисел в последовательности нет.

7. Дана последовательность из 15 целых чисел. Найдите наибольшее и наименьшее числа в этой последовательности и поменяйте эти числа местами.

8. Дана последовательность из 15 вещественных чисел. Поменяйте наименьшее число в этой последовательности с первым числом и наибольшее число с последним числом.

9. Дан список вещественных чисел, содержащий 10 элементов. Если минимальный элемент или максимальный элемент этого списка равен 3.5, то поменяйте эти элементы местами.

10. Дан список целых чисел, содержащий 10 элементов. Если минимальный элемент в этом списке находится на первом месте, а максимальный – на последнем месте, то поменяйте местами эти элементы.

11. Дан список из 15 вещественных чисел. Если минимальный элемент списка предшествует максимальному элементу, то подсчитайте сумму максимального и минимального элементов, иначе – вычислите разность.

12. Дан список из 15 вещественных чисел. Если максимальный элемент списка предшествует минимальному элементу, то подсчитайте произведение минимального и максимального элементов, иначе – среднее арифметическое.

13. Дан список b из 15 целых чисел. Найдите наименьшее из чисел, расположенных от начала списка до максимального элемента списка b .

14. Дан список из 10 целых чисел. Если минимальный элемент списка расположен на 4 месте, а максимальный – рядом (слева или справа), то подсчитайте среднее арифметическое максимального и минимального элементов списка.

15. Дан список из 15 целых чисел. Если минимальный элемент списка меньше максимального в 3 раза, то каждый элемент списка увеличьте в 3 раза.

16. Дан список b из 15 целых чисел. Найдите наибольшее из чисел, расположенных от минимального элемента до конца списка b .

17. Дан список из 15 целых чисел. Если минимальный или максимальный элемент равен 0, то обнулите элементы списка, предшествующие минимальному

элементу в списке.

18. Дан список из 15 целых чисел. Если минимальный элемент меньше максимального в 2 раза, то все элементы, предшествующие максимальному элементу, установите равными минимальному элементу.

19. Дан список из 10 целых чисел. Если максимальный элемент находится на 2 месте, а минимальный элемент – на 5 месте, то все элементы, расположенные от минимального элемента и до конца списка, установите равными максимальному элементу.

20. Даны натуральное число n и целочисленный список $a[15]$. Определите среднее арифметическое тех чисел, которые кратны n .

21. Дан целочисленный список $a[15]$. Найдите наименьшее из чисел, значения которых кратны 3, в последовательности чисел $a[15]$. Предусмотрите ситуацию, когда таких чисел в последовательности нет.

22. Дан целочисленный список $a[10]$. Найдите наибольшее из чисел, значения которых кратны 5 и не кратны 7, в последовательности чисел $a[10]$. Предусмотрите ситуацию, когда таких чисел в последовательности нет.

23. Дан список b из 15 целых чисел. Найдите наибольшее из чисел, расположенных от начала списка до минимального элемента списка b .

24. Дан список из 10 вещественных чисел. Максимальный элемент списка увеличьте в 2 раза, а минимальный элемент уменьшите в 2 раза. Выведите преобразованный список.

25. Дан список из 15 вещественных чисел. Максимальный элемент списка замените суммой минимального и максимального элементов, а минимальный элемент – разностью максимального и минимального элементов. Выведите преобразованный список.

26. Дан список из 10 целых чисел, среди которых могут быть одинаковые числа. Установите все элементы, имеющие наименьшее значение, равными среднему арифметическому значению положительных элементов списка. Предусмотрите ситуацию, когда в списке положительных чисел нет.

27. Дан список из 15 целых чисел, значения которых лежат в диапазоне от 1 до 1000. Определите индекс первого элемента тройки элементов списка, сумма которых максимальна. Например, 1, 2 и 3 элементы, 4, 5 и 6 элементы и т.д.

28. Дан список из 20 целых чисел. Определите индекс первого элемента пары элементов списка, сумма которых максимальна. Например, 1 и 2 элементы, 3 и 4 элементы и т.д.

29. Дан список из 20 целых чисел. Определите индекс первого элемента пары элементов списка, произведение которых минимально. Например, 1 и 2 элементы, 3 и 4 элементы и т.д.

30. Дан список из 15 целых чисел, значения которых лежат в диапазоне от 1 до 1000. Определите индекс первого элемента тройки элементов списка, сумма которых минимальна. Например, 1, 2 и 3 элементы, 4, 5 и 6 элементы и т.д.

Задачи второго уровня сложности.

1. Даны натуральное число n и целые числа a_1, \dots, a_n . Подсчитать максимальную длину последовательности подряд расположенных элементов, имеющих одинаковые значения.
2. Даны натуральное число n и целые числа a_1, \dots, a_n . Подсчитать максимальную длину последовательности подряд расположенных элементов, каждый из которых больше предыдущего.
3. Даны натуральное число n и целочисленный список $a[n]$, среди элементов которого есть одинаковые. Создайте список b из различных элементов списка a .
4. Даны натуральное число n и целые числа a_1, \dots, a_n , среди которых могут быть как положительные, так и отрицательные числа. Создайте список b , в начале которого поместить отрицательные числа, а в конце – положительные.
5. Даны натуральное число n и символьный список $a[n]$, в котором хранятся латинские буквы и арабские цифры. Создайте символьный список b , поместив в него только латинские буквы из списка a .
6. Даны натуральное число n и символьный список $a[n]$, в котором хранятся латинские буквы и арабские цифры. Создайте целочисленный список b , поместив в него только цифры из списка a .
7. Даны натуральное число n и целочисленный список $a[n]$. Упорядочьте список a по возрастанию.
8. Даны натуральное число n и целочисленный список $a[n]$. Упорядочьте список a по убыванию.
9. Даны натуральное число n и символьный список $a[n]$, в котором хранятся латинские буквы и арабские цифры. Создайте символьный список b , поместив в начале которого числа, а в конце – латинские буквы из списка a .
10. Даны натуральное число n и целочисленный список $a[n]$, состоящий только из 0 и 1. Определите самое большое количество подряд идущих единиц и выведите на экран индексы начала и конца этого диапазона.
11. Даны натуральное число n и целочисленный список $a[n]$, состоящий только из 0 и 1. Определите самое большое количество подряд идущих нулей и выведите на экран индекс начала и количество нулей самой длинной серии нулей.
12. Даны натуральное число n и целочисленный список $a[n]$. Выведите значение *true* для случая, если список $a[n]$ представляет знакопеременную последовательность, и *false* – в противном случае. Последовательность является знакопеременной, если знак очередного элемента противоположен знакам его соседей.
13. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$, упорядоченные по возрастанию (предыдущий элемент меньше последующего). Требуется получить третий упорядоченный по возрастанию список путем слияния первых двух.

14. Даны натуральное число n и целочисленный список $a[n]$. Выведите значение *true* для случая, если список $a[n]$ представляет возрастающую или убывающую последовательность, и *false* – в противном случае.

15. Даны натуральное число n и целые числа a_1, \dots, a_n . Подсчитайте максимальную длину последовательности подряд расположенных элементов, имеющих отрицательные значения. Возможно, что таких чисел в списке a нет.

16. Даны натуральное число n и целые числа a_1, \dots, a_n . Подсчитайте максимальную длину последовательности подряд расположенных элементов, имеющих положительные значения. Возможно, что таких чисел в списке a нет.

17. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которых нет в списке b . Возможно, что таких чисел в списке a и в списке b нет.

18. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые есть в списке b . Возможно, что таких чисел в списке a и в списке b нет.

19. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые встречаются в списке b более одного раза. Возможно, что таких чисел в списке a и в списке b нет.

20. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые больше минимального элемента среди четных чисел списка b . Возможно, что таких чисел в списке a и в списке b нет.

21. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые меньше максимального элемента среди нечетных чисел списка b . Возможно, что таких чисел в списке a и в списке b нет.

22. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые меньше максимального элемента среди чисел списка b , значения которых кратны 3. Возможно, что таких чисел в списке a и в списке b нет.

23. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые больше минимального элемента среди чисел списка b , значения которых не кратны 7. Возможно, что таких чисел в списке a и в списке b нет.

24. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые больше суммы минимального и максимального элементов среди чисел списка b , значения которых кратны 5. Возможно, что таких чисел в списке a и в списке b нет.

25. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые меньше произведения минимального и максимального элементов среди нечетных чисел списка b . Возможно, что таких чисел в списке a и в списке b нет.

26. Даны натуральное число n и целочисленные списки $a[n]$ и $b[n]$. Из списка a выберите элементы, которые меньше среднего арифметического минимального и

максимального элементов среди четных чисел списка b . Возможно, что таких чисел в списке a и в списке b нет.

27. Даны натуральное число n и целочисленный список $a[n]$, элементами которого являются двузначные числа. Из списка a выберите элементы, у которых старшая цифра ровно в 2 раза меньше младшей цифры. Возможно, что таких чисел в списке a нет.

28. Даны натуральное число n и целочисленный список $a[n]$, элементами которого являются двузначные числа. Из списка a выберите элементы, значения которых не принадлежат диапазону $[50, 60]$. Возможно, что таких чисел в списке a нет.

29. Даны натуральное число n и символьный список $a[n]$, в котором хранятся буквы русского алфавита. Создайте символьный список b , поместив в него буквы из списка a , лежащие в диапазоне $[б, м]$. Возможно, что таких букв в списке a нет.

30. Даны натуральное число n и символьный список $a[n]$, в котором хранятся буквы латинского алфавита. Создайте символьный список b , поместив в него буквы из списка a , не лежащие в диапазоне $[d, p]$. Возможно, что таких букв в списке a нет.

2.5.2. Задачи на восходящую рекурсию

Задачи первого уровня сложности.

1. Дана вещественная матрица размером 4 строки, 5 столбцов. Переставляя ее строки и столбцы, добейтесь того, чтобы наибольший элемент (один из них) оказался в верхнем левом углу.

2. Определите, является ли заданная целочисленная квадратная матрица порядка 5 симметричной относительно главной диагонали.

3. Дана вещественная матрица размером 4 строки, 5 столбцов. Поменяйте местами максимальный и минимальный элементы матрицы.

4. Дана целочисленная квадратная матрица порядка 5. Найдите максимальный элемент среди элементов, лежащих ниже главной диагонали, и максимальный элемент среди элементов, лежащих выше главной диагонали, поменяйте их местами.

5. Дана целочисленная квадратная матрица порядка 5. Найдите максимальный элемент среди элементов, лежащих левее вспомогательной диагонали, и максимальный элемент среди элементов, лежащих правее вспомогательной диагонали, поменяйте их местами.

6. Среди строк целочисленной квадратной матрицы порядка 5 найдите строку с минимальной суммой элементов.

7. Дана целочисленная квадратная матрица порядка 5. Найдите минимальный элемент среди элементов, лежащих ниже главной диагонали, и максимальный

элемент среди элементов, лежащих выше главной диагонали, вычислите их среднее арифметическое.

8. Дана целочисленная квадратная матрица порядка 5. Найдите минимальный элемент среди элементов, лежащих левее вспомогательной диагонали, и максимальный элемент среди элементов, лежащих правее вспомогательной диагонали, и вычислить их среднее геометрическое.

9. Среди столбцов целочисленной квадратной матрицы порядка 5 найдите столбец с максимальной суммой элементов.

10. Среди тех столбцов целочисленной матрицы размером 3 строки, 5 столбцов, которые содержат только такие элементы, значения которых по модулю не превышают 10, найдите столбец с минимальным произведением элементов.

11. Даны целые числа a_1, \dots, a_{10} , целочисленная квадратная матрица порядка 5. Замените нулями в матрице те элементы, для которых имеются равные числа среди a_1, \dots, a_{10} .

12. В двумерном целочисленном массиве размером 4 строки, 5 столбцов поменяйте местами строки, симметричные относительно середины массива (горизонтальной линии).

13. В двумерном целочисленном массиве размером 4 строки, 5 столбцов поменяйте местами столбцы, симметричные относительно середины массива (вертикальной линии).

14. Даны две вещественные квадратные матрицы порядка 4. Получите новую матрицу прибавлением к элементам каждого столбца первой матрицы минимального элемента соответствующего столбца второй матрицы.

15. В целочисленной квадратной матрице порядка 4 найдите наибольший по модулю элемент. Получите квадратную матрицу порядка 3 путем выбрасывания из исходной матрицы строки и столбца, на пересечении которых расположен элемент с найденным значением.

16. В данной вещественной квадратной матрице порядка 5 найдите наименьший элемент. Получите квадратную матрицу порядка 4 путем выбрасывания из исходной матрицы строки и столбца, на пересечении которых расположен элемент с найденным значением.

17. Дана вещественная матрица размером 4 строки, 5 столбцов. Получите новую матрицу путем вычитания из всех элементов данной матрицы наименьшего по модулю элемента.

18. Дана целочисленная квадратная матрица порядка 4. Найдите в каждой строке наибольший элемент и поменяйте его местами с элементом, расположенным на главной диагонали.

19. Дана целочисленная квадратная матрица порядка 5. Найдите в каждой строке наименьший элемент и поменяйте его местами с элементом, расположенным на вспомогательной диагонали.

20. Дана вещественная матрица размером 4 строки, 5 столбцов. Определите числа b_1, \dots, b_4 , равные значениям средних арифметических элементов строк.

21. Дана вещественная матрица размером 4 строки, 5 столбцов. Определите числа b_1, \dots, b_5 , равные значениям средних арифметических элементов столбцов.

22. Дана вещественная матрица размером 4 строки, 5 столбцов. Определите числа b_1, \dots, b_5 , равные среднему арифметическому значению максимального и минимального элементов каждого столбца.

23. В данной вещественной квадратной матрице порядка 5 найдите сумму элементов строки, в которой расположен элемент с наименьшим значением. Предполагается, что такой элемент единственный.

24. В данной вещественной квадратной матрице порядка 5 найдите среднее геометрическое положительных элементов столбца, в котором расположен элемент с наименьшим значением. Предполагается, что такой элемент единственный.

25. Дана целочисленная квадратная матрица порядка 4. Для каждой строки найдите минимальный элемент или максимальный в зависимости от условия: если на главной диагонали находится элемент больше элемента, расположенного на вспомогательной диагонали, то найдите минимальный элемент, иначе – максимальный.

26. Дана целочисленная квадратная матрица порядка 4. Ниже главной диагонали найдите минимальный элемент среди четных элементов, а выше главной диагонали – среди нечетных элементов.

27. Дана целочисленная квадратная матрица порядка 4. Ниже главной диагонали найдите минимальный элемент среди положительных элементов, а выше главной диагонали – максимальный элемент среди отрицательных элементов.

28. Дана целочисленная квадратная матрица порядка 5. Левее вспомогательной диагонали найдите минимальный элемент среди четных элементов, а правее вспомогательной диагонали – среди нечетных элементов.

29. Дана целочисленная квадратная матрица порядка 5. Левее вспомогательной диагонали найдите минимальный элемент среди положительных

элементов, а правее вспомогательной диагонали – максимальный элемент среди отрицательных элементов.

30. Дана целочисленная квадратная матрица порядка 5. Для каждого столбца найдите минимальный элемент или максимальный в зависимости от условия: если на главной диагонали находится элемент меньше элемента, расположенного на вспомогательной диагонали, то найдите минимальный элемент, иначе – максимальный.

Задачи второго уровня сложности.

1. Дана вещественная матрица размером n строк, m столбцов. Упорядочьте ее строки по возрастанию их первых элементов.

2. Дана вещественная матрица размером n строк, m столбцов. Упорядочьте ее строки по возрастанию значений элементов.

3. Дана вещественная матрица размером n строк, m столбцов. Для каждой строки найдите наибольший элемент и упорядочьте строки матрицы по возрастанию значений наибольших элементов.

4. Дана вещественная матрица размером n строк, m столбцов. Для каждой строки найдите наименьший элемент и упорядочьте строки матрицы по возрастанию значений наименьших элементов.

5. Дана вещественная матрица размером n строк, m столбцов. Для каждой строки найдите сумму элементов и упорядочьте строки матрицы по возрастанию сумм.

6. Дана вещественная матрица размером n строк, m столбцов. Для каждого столбца найдите сумму элементов и упорядочьте столбцы матрицы по возрастанию сумм.

7. Дана целочисленная матрица размером n строк, m столбцов. Для каждого столбца найдите сумму нечетных элементов и упорядочьте столбцы матрицы по возрастанию сумм.

8. Дана целочисленная матрица размером n строк, m столбцов. Для каждого столбца найдите сумму положительных элементов и упорядочьте столбцы матрицы по возрастанию сумм.

9. Дана целочисленная матрица размером n строк, m столбцов. Для каждого столбца найдите произведение элементов, значения которых лежат в диапазоне $[1, 10]$, упорядочьте столбцы матрицы по убыванию произведений.

10. Дана целочисленная матрица размером n строк, m столбцов. Для каждого столбца найдите наибольшие элементы и упорядочьте столбцы матрицы по убыванию наибольших элементов.

11. Дана вещественная матрица размером n строк, m столбцов. Упорядочьте ее столбцы по возрастанию их первых элементов.

12. Дана целочисленная матрица размером n строк, m столбцов. Найдите и выведите строку, в которой длина максимальной серии одинаковых элементов максимальна.

13. Дана целочисленная матрица размером n строк, m столбцов. Найдите и выведите столбец, в котором длина максимальной серии возрастающих по значению элементов максимальна.

14. Дана вещественная матрица размером n строк, m столбцов. Для каждого столбца найдите среднее арифметическое элементов и упорядочьте столбцы матрицы по возрастанию среднего арифметического элементов.

15. Дана вещественная матрица размером n строк, m столбцов. Для каждой строки найдите среднее арифметическое элементов и упорядочьте строки матрицы по возрастанию среднего арифметического элементов.

16. Дана вещественная матрица размером n строк, m столбцов. Для каждого столбца найдите среднее геометрическое положительных элементов и упорядочьте столбцы матрицы по возрастанию среднего геометрического элементов.

17. Дана вещественная матрица размером n строк, m столбцов. Для каждой строки найдите среднее геометрическое положительных элементов и упорядочьте строки матрицы по убыванию среднего геометрического элементов.

18. Дана вещественная матрица размером n строк, m столбцов. Найдите строку, содержащую максимальный элемент, и упорядочьте эту строки матрицы по возрастанию значений элементов.

19. Дана вещественная матрица размером n строк, m столбцов. Найдите строку, содержащую минимальный элемент, и упорядочьте эту строки матрицы по убыванию значений элементов.

20. Дана вещественная матрица размером n строк, m столбцов. Найдите столбец, содержащий минимальный элемент, и упорядочьте этот столбец матрицы по убыванию значений элементов.

21. Дана вещественная матрица размером n строк, m столбцов. Найдите столбец, содержащий максимальный элемент, и упорядочьте этот столбец матрицы по возрастанию значений элементов.

22. Дана целочисленная матрица размером n строк, m столбцов. Найдите столбец, сумма элементов которого минимальна, и упорядочьте этот столбец матрицы по убыванию значений элементов.

23. Дана целочисленная матрица размером n строк, m столбцов. Найдите строку, сумма элементов которой максимальна, и упорядочьте эту строку матрицы по убыванию значений элементов.

24. Дана целочисленная матрица размером n строк, m столбцов. Найдите строку, среднее арифметическое элементов которой максимально, и упорядочьте эту строку матрицы по возрастанию значений элементов.

25. Дана целочисленная матрица размером n строк, m столбцов. Найдите столбец, среднее арифметическое элементов которого максимально, и упорядочьте этот столбец матрицы по возрастанию значений элементов.

26. Дана целочисленная матрица размером n строк, m столбцов. Найдите строку, среднее геометрическое элементов которой максимально, и упорядочьте эту строку матрицы по убыванию значений элементов.

27. Дана целочисленная матрица размером n строк, m столбцов. Найдите столбец, среднее геометрическое элементов которого минимально, и упорядочьте этот столбец матрицы по убыванию значений элементов.

28. Дана целочисленная матрица размером n строк, m столбцов. Найдите строку, произведение элементов которой максимально, и упорядочьте эту строку матрицы по убыванию значений элементов.

29. Дана целочисленная матрица размером n строк, m столбцов. Найдите столбец, произведение элементов которого минимально, и упорядочьте этот столбец матрицы по убыванию значений элементов.

30. Дана целочисленная матрица размером n строк, m столбцов. Найдите строку, количество различных элементов в которой максимально, и упорядочьте эту строку матрицы по возрастанию значений элементов.

3. СОДЕРЖАНИЕ ОТЧЕТА

Оформление решения задач по индивидуальным заданиям, так как показано в приложении.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ К ЗАЩИТЕ ЛАБОРАТОРНОЙ РАБОТЫ

Какая из рекурсий работает быстрее? Нисходящая или восходящая?

Чему равен результат предиката в терминальном случае при восходящей рекурсии?

Чему равен результат предиката в терминальном случае при нисходящей рекурсии?

Что является необходимым и достаточным условием для записи рекурсивных определений предикатов на Прологе?

Приложение

Вариант оформления индивидуального задания

1.1 Реализация задачи первого уровня сложности

1.1.1 Спецификация

Функция возвращает матрицу, в которой строки упорядочены по возрастанию суммы их элементов.

1.1.2 Структура входных данных

Входными данными является целочисленная матрица (двумерный список) `intList`.

1.1.3 Структура выходных данных

Выходными данными является целочисленная матрица.

1.1.4 Декларативное определение результата

Результатом применения функции является преобразованная матрица, в которой строки упорядочены по возрастанию суммы элементов.

1.1.5 Функциональные тесты

В таблице 1 представлены тестовые случаи для функции.

Таблица 1 - Тестовые случаи для функции

Проверяемая ситуация	Вызов функции со входными значениями	Предполагаемый результат
Пустая матрица	lab5.1 ([], Matrix)	No Solution
Целочисленная матрица	lab5.1 ([[5, -4, 2], [2, 1, -4], [-9, 0, 0]], Matrix)	Matrix=[[-9, 0, 0], [2, 1, -4], [5, -4, 2]]

1.1.6 Графическое решение

На рисунке 1 представлено графическое решение.

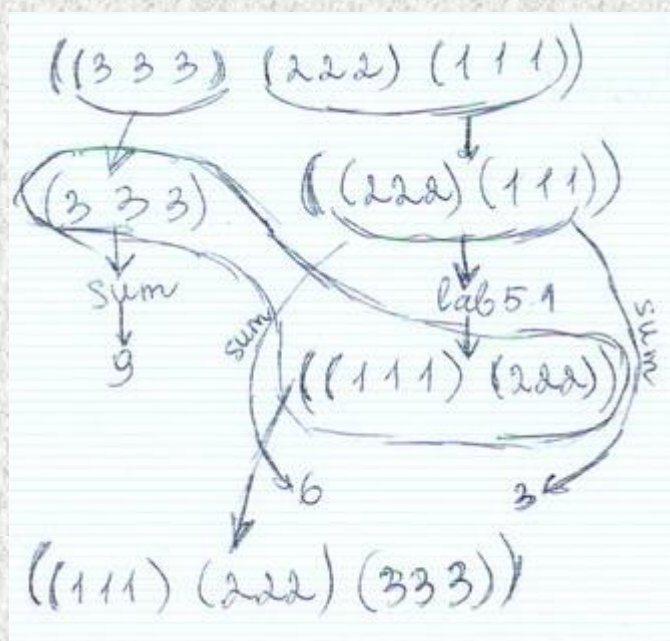


Рисунок 1 – Графическое решение для функции

1.1.7 Текст программы на языке Prolog

На рисунке 2 представлен текст программы на языке Prolog.


```

domains
    intList = integer*
    intMatrix = intList*

predicates
    lab5.1(intMatrix, intMatrix)
    insert_row(intMatrix, intList, intMatrix)
    sum(intList, integer)

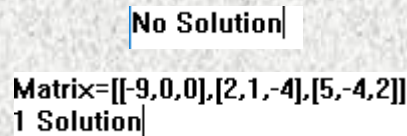
clauses
/* Функция возвращает матрицу, в которой строки упорядочены
по возрастанию суммы их элементов. */
lab5.1([H], [H]).
lab5.1([H|T], Matrix):-
    lab5.1(T, SortTailMatrix),
    insert_row(SortTailMatrix, H, Matrix).
/* Функция сортирует строки по их сумме */
insert_row(Matrix, [], Matrix).
insert_row([], Row, [Row]).
insert_row([H|T], Row, SortMatrix) :- sum(Row, SumRow), sum(H,
SumHead), SumRow < SumHead,
    SortMatrix = [Row | [H|T]].
insert_row([H|T], Row, SortMatrix) :- sum(Row, SumRow), sum(H,
SumHead), SumRow >= SumHead, insert_row(T, Row, SortTail),
SortMatrix = [H | SortTail].
/* Функция возвращает сумму элементов строки */
sum([], 0).
sum([H|T], Sum) :-

```

Рисунок 2 – Текст программы

1.1.8 Результаты теста

На рисунке 3 представлен результат работы тестового драйвера для данной функции.



The image shows a screenshot of a test result. At the top, it says "No Solution" in a blue box. Below that, it shows a matrix: **Matrix=[[-9,0,0],[2,1,-4],[5,-4,2]]**. At the bottom, it says "1 Solution" in a blue box.

Рисунок 3 – Результаты тестирования функции

1.2 Реализация задачи второго уровня сложности

1.2.1 Спецификация

Функция совершает подсчет максимальной длину последовательности подряд расположенных элементов, имеющих положительные значения. Возможно, что таких чисел в массиве нет.

1.2.2 Структура входных данных

Входными данными являются список List.

1.2.3 Структура выходных данных

Выходными данными является переменная целочисленного типа.

1.2.4 Декларативное определение результата

Результатом применения функции является преобразованная матрица, в которой строки упорядочены по возрастанию суммы элементов.

1.2.5 Функциональные тесты

В таблице 2 представлены тестовые случаи для функции.

Таблица 2 - Тестовые случаи для функции

Проверяемая ситуация	Вызов функции со входными значениями	Предполагаемый результат
Последовательность с подряд идущими положительными числами	lab5.2 (-3, -6, 5, 9, -4, -3)	2
Последовательность положительных чисел	lab5.2 (5, 7, 4, 3, 9, 1, 2)	7
Последовательность отрицательных чисел	lab5.2 (-1, -9, -3, -4, -6)	0
Знакопеременная последовательность	lab5.2 (-1, 6, -9, 6, -2, 4)	1
Пустая последовательность	lab5.2 ()	0

1.2.6 Графическое решение

На рисунке 4 представлено графическое решение.

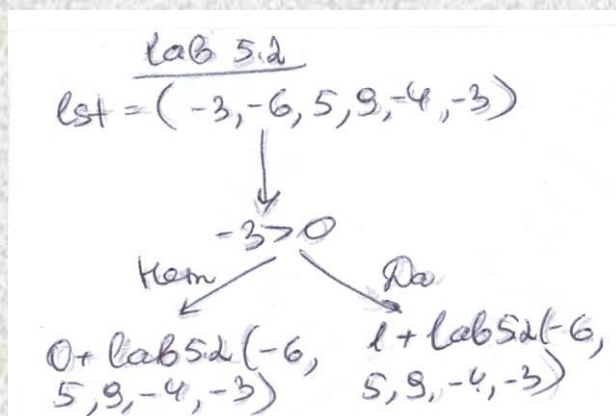


Рисунок 4 – Графическое решение для функции

1.2.7 Текст программы на языке Prolog

На рисунке 5 представлен текст программы на языке Prolog.

```
domains
    List = integer*
predicates
    lab5.2(List, integer, integer)
    listRowLength(List, integer)
clauses
    /* Функция возвращает результат */
    lab5.2([H|T], N, Tail):- H>=0,
                                lab5.2(T, Res, Tail),
                                N=Res+1.
    lab5.2([H|T], 0, [H|T]):-H<0.
    lab5.2([], 0, []).
    /* Функция совершает подсчет максимальной длины
    последовательности подряд расположенных элементов,
    имеющих положительные значения. Возможно, что таких чисел в
    массиве нет. */
    listRowLength([H|T], MAX):-H>=0,
                                lab5.2([H|T], N, SP),
                                listRowLength(SP, MX),
                                N>MX,
                                Max=N.
    listRowLength([H|T], MAX):-H<0,
                                ListRowLength(T, MAX)
```

Рисунок 5 – Текст программы

1.2.8 Результаты теста

На рисунке 6 представлен результат работы тестового драйвера для данной функции.

```
listRowLength() :- listRowLength([-1, -2, 3, 4, -5, 6], 2),
                  listRowLength([1, 2, 3, 4, 5], 5),
                  listRowLength([-1, -2, -3, -4, -5], 0),
                  listRowLength([-1, 2, -3, 4, -5, 1], 1),
                  listRowLength([], 0).

yes
```

Рисунок 6 – Результаты тестирования функции

2 Лабораторная работа «Программирование с использованием двумерных массивов» на языке Prolog

2.1 Реализация задачи первого уровня сложности

2.1.1 Спецификация

Функция возвращает список, в котором минимальный элемент заменен на целую часть среднего арифметического всех элементов списка.

2.1.2 Структура входных данных

Входными данными является список целых чисел `intList`.

2.1.3 Структура выходных данных

Выходными данными является список целых чисел.

2.1.4 Декларативное определение результата

Результатом применения функции является преобразованный список, в котором минимальный элемент заменен на целую часть среднего арифметического всех элементов.

2.1.5 Функциональные тесты

В таблице 3 представлены тестовые случаи для функции.

Таблица 3 - Тестовые случаи для функции

Проверяемая ситуация	Вызов функции со входными значениями	Предполагаемый результат
Пустой список	lab6.1([], Out)	No Solution
Список целых чисел	Lab6.1([5, 1, 4, -6, 5, 2, 8], Out)	Out=[5, 1, 4, 2, 5, 2, 8]

2.1.6 Графическое решение

На рисунке 7 представлено графическое решение.

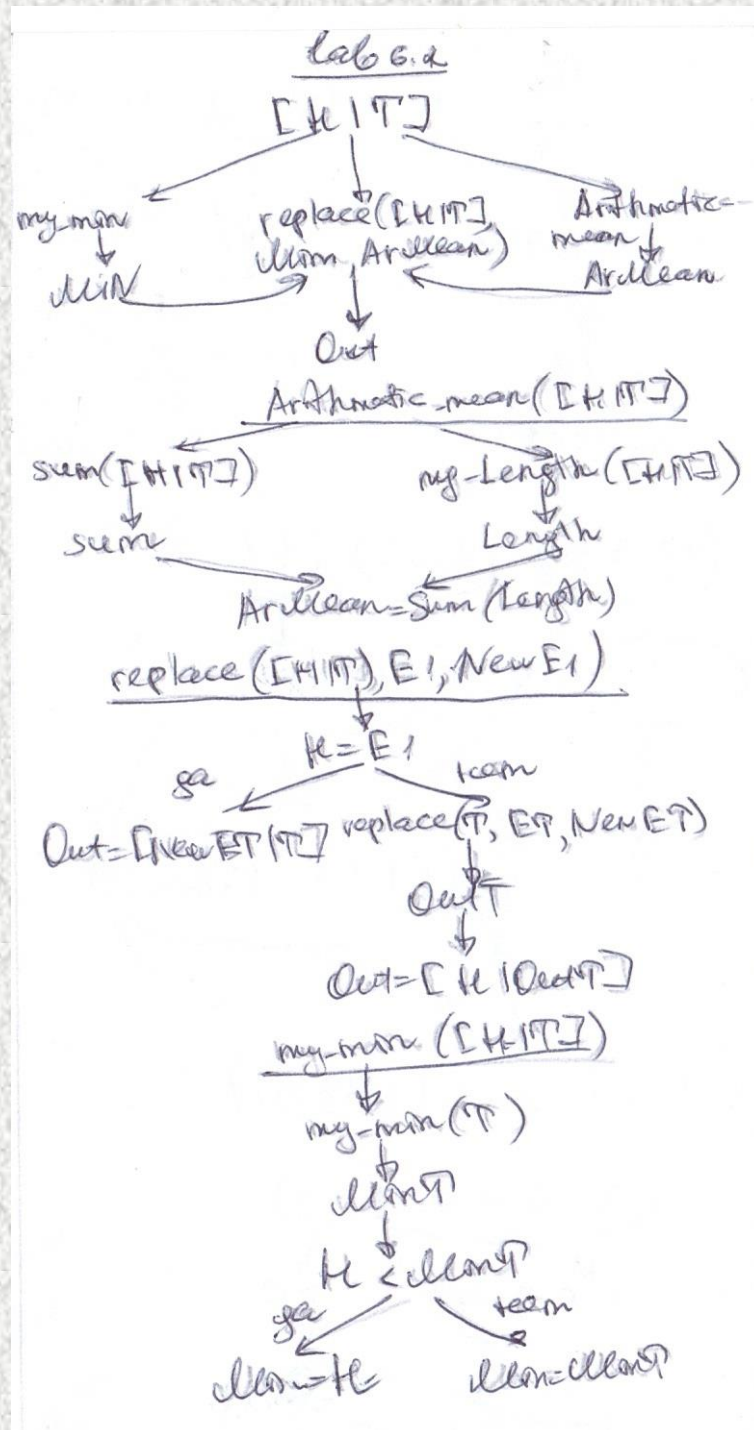


Рисунок 7 – Графическое решение для функции

2.1.7 Текст программы на языке Prolog

На рисунках 8-9 представлен текст программы на языке Lisp.

```
domains
    intList = integer*
predicates
    lab6.1(intList, intList)
    my_min(intList, integer)
    my_length(intList, integer)
    sum(intList, integer)
    arithmetic_mean(intList, real)
    replace(intList, integer, integer, intList)
clauses
/* Функция возвращает результат */
lab6.1([H|T], Out) :-
    my_min([H|T], Min), /* Find minimum */
    arithmetic_mean([H|T], ArMean), /* Find arithmetic mean */
    replace([H|T], Min, ArMean, Out). /* Replace */
/* Функция возвращает минимальный элемент списка */
my_min([H|T], Min) :-
    my_min(T, MinT),
    H < MinT,
    Min = H.
```

Рисунок 8 – Текст программы


```

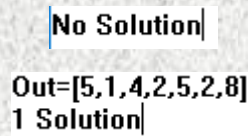
my_min([H|T], Min) :- my_min(T, MinT),
    H > MinT,
    Min = MinT.
my_min([Min], Min).
/* Функция возвращает длину списка */
my_length([H|T], Length) :-
    my_length(T, LengthT),
    Length = LengthT + 1.
my_length([X], 1).
/* Функция возвращает сумму элементов списка */
sum([H|T], Sum) :-
    sum(T, SumT),
    Sum = H + SumT.
sum([X], X).
/* Функция возвращает целую часть среднего арифметического всех
элементов списка */
arithmetic_mean([H|T], ArMean) :-
    sum([H|T], Sum),
    my_length([H|T], Length),
    ArMean = Sum div Length.
/* Функция заменяет число на среднее арифметическое */
replace([H|T], El, NewEl, Out) :-
    H = El,
    Out = [NewEl|T].
replace([H|T], El, NewEl, Out) :-
    H <> El,
    replace(T, El, NewEl, OutT),
    Out = [H|OutT].
replace([], El, NewEl, [])

```

Рисунок 9 – Текст программы

2.1.8 Результаты теста

На рисунке 10 представлен результат работы тестового драйвера для данной функции.



The image shows a screenshot of a test result interface. It contains three lines of text: 'No Solution' in a blue box, 'Out=[5,1,4,2,5,2,8]' in a black box, and '1 Solution' in a green box.

```
No Solution|  
Out=[5,1,4,2,5,2,8]  
1 Solution|
```

Рисунок 10 – Результаты тестирования функции