

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. Р.Е. АЛЕКСЕЕВА»

Ю.П. ЕМЕЛЬЯНОВА, П.В. ПАКШИН

ПРОГРАММИРОВАНИЕ В SCILAB

*Рекомендовано Ученым советом Нижегородского государственного
технического университета им. Р.Е. Алексеева в качестве учебного пособия
(лабораторного практикума) для студентов, обучающихся по направлениям
подготовки 231300 «Прикладная математика»,
и 230400 «Информационные системы и технологии»*

Нижний Новгород 2015

УДК 004
ББК 32.973
Е 601

Р е ц е н з е н т ы:

доктор физико-математических наук, профессор *Д.В. Баландин*;
кандидат физико-математических наук, доцент *Л.В. Широков*

Емельянова Ю.П., Пакшин П.В.

Е 601 Программирование в Scilab: учеб. пособие (лабораторный практикум) / Ю.П. Емельянова, П.В. Пакшин; Нижегород. гос. техн. ун-т им. Р.Е. Алексеева. – Нижний Новгород, 2015. – 114 с.

ISBN 978-5-502-00666-8

Посвящен основам программирования с использованием пакета Scilab. Изложены базовые основополагающие темы программирования начального уровня и содержатся основные сведения и практические рекомендации по работе с пакетом Scilab.

Адресован студентам младших курсов высших учебных заведений, обучающимся по направлениям подготовки 231300 «Прикладная математика» и 230400 «Информационные системы и технологии» в рамках дисциплин «Программирование для ЭВМ», «Программные и аппаратные средства информатики», а также студентам начальных курсов других математических и инженерных специальностей в рамках дисциплин «Информатика», «Информационные технологии» и может быть интересен начинающим программистам.

Рис. 14. Табл. 2. Библиогр.: 5 назв.

УДК 004
ББК 32.973

ISBN 978-5-502-00666-8

© Нижегородский государственный
технический университет
им. Р.Е. Алексеева, 2015
© Емельянова Ю.П.,
Пакшин П.В., 2015

СОДЕРЖАНИЕ

ПРЕДИСЛОВИЕ	4
ВВЕДЕНИЕ	5
ЛАБОРАТОРНЫЕ РАБОТЫ	6
1. ПРОГРАММИРОВАНИЕ ЗАДАЧ ЛИНЕЙНОЙ СТРУКТУРЫ	6
2. ПРОГРАММИРОВАНИЕ ЗАДАЧ РАЗВЕТВЛЯЮЩЕЙСЯ СТРУКТУРЫ.....	16
3. ПРОГРАММИРОВАНИЕ ЗАДАЧ ЦИКЛИЧЕСКОЙ СТРУКТУРЫ.....	33
4. ПРОГРАММИРОВАНИЕ ЗАДАЧ СЛОЖНОЙ ЦИКЛИЧЕСКОЙ СТРУКТУРЫ.....	46
5. ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ ОДНОМЕРНЫХ МАССИВОВ. ФУНКЦИИ	55
6. ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ ДВУМЕРНЫХ МАССИВОВ. ФУНКЦИИ».....	73
7. ПРОГРАММИРОВАНИЕ В SCILAB. ИТОГОВАЯ РАБОТА.....	96
СПИСОК ЛИТЕРАТУРЫ	99
ПРИЛОЖЕНИЯ	100
Приложение А. Установка Scilab на ПК	100
Приложение Б. Основные элементы блок-схемы.....	106
Приложение В. Образец оформления лабораторной работы	107
Приложение Г. Ответы к задачам для самостоятельного решения.....	109
Ответы к лабораторной работе 1.....	109
Ответы к лабораторной работе 2.....	109
Ответы к лабораторной работе 3.....	110
Ответы к лабораторной работе 5.....	110
Ответы к лабораторной работе 6.....	111
Ответы к лабораторной работе 7.....	113

ПРЕДИСЛОВИЕ

В лабораторном практикуме изложены базовые основополагающие темы программирования начального уровня и на практических примерах рассмотрены основные принципы работы в системе Scilab. Назначение данного практикума – закрепление пройденного теоретического курса и приобретение практических навыков программирования начального уровня, а также навыков работы в среде Scilab.

Практикум состоит из семи отдельных лабораторных работ и четырех приложений.

Лабораторная работа 1 посвящена программированию задач линейной структуры и овладению основными простейшими навыками по установке и работе в среде Scilab.

В лабораторной работе 2 рассматриваются задачи разветвляющейся структуры и изучаются основы работы с условным оператором.

В лабораторных работах 3 и 4 решаются задачи циклической структуры, изучаются операторы цикла, реализующие простые и вложенные циклические процессы.

Лабораторная работа 5 посвящена программированию задач с использованием одномерных массивов и приобретению навыков работы в написании и отладке программ с использованием функций.

Лабораторная работа 6 посвящена программированию задач с использованием двумерных массивов и функций.

Лабораторная работа 7 – итоговая работа, где приведены задачи для самостоятельного решения, с целью рубежного контроля за успеваемостью студента.

В приложениях даны указания по установке пакета Scilab на ПК, а также справочник по основным элементам блок-схем, представлен образец оформления лабораторной работы, приведены ответы к задачам для самостоятельного решения.

Практикум адресован студентам младших курсов высших учебных заведений, обучающимся по направлениям подготовки 231300 «Прикладная математика» и 230400 «Информационные системы и технологии» в рамках дисциплин «Программирование для ЭВМ», «Программные и аппаратные средства информатики», а также студентам начальных курсов других математических и инженерных специальностей в рамках дисциплин «Информатика», «Информационные технологии» и может быть интересен начинающим программистам.

ВВЕДЕНИЕ

Scilab – пакет прикладных математических программ, представляющий собой мощное открытое окружение для инженерных и научных расчётов. Пакет Scilab - свободно распространяемая бесплатная программа. Доступны версии Scilab для различных операционных систем: Linux, Windows, MacOS. Последняя версия Scilab всегда доступна на официальном сайте www.scilab.org.

Scilab был создан в 80-х гг. XX века группой исследователей из Национального исследовательского института INRIA (Франция). С 1994 года Scilab распространяется вместе с исходным кодом через Интернет.

По своим возможностям пакет Scilab – самая полная доступная альтернатива MATLAB. Пакет содержит сотни встроенных математических функций и команд, но также существует возможность добавления новых собственных функций и приложений, написанных на различных языках (C, C++, Fortran, MATLAB ит.д.). Также имеются разнообразные структуры данных (списки, полиномы, рациональные функции, линейные системы), интерпретатор и достаточно мощный собственный язык программирования высокого уровня. Таким образом, в Scilab доступно выполнение различных инженерных и научных вычислений, таких как:

- решение нелинейных уравнений и систем;
- решение задач линейной алгебры;
- решение задач оптимизации;
- дифференцирование и интегрирование;
- обработка экспериментальных данных (интерполяция и аппроксимация, метод наименьших квадратов);
- решение обыкновенных дифференциальных уравнений и систем;
- scicos – инструмент для редактирования блочных диаграмм и симуляции;
- 2D- и 3D-графика и анимация;
- обработка сигналов;
- параллельная работа;
- статистика.

ЛАБОРАТОРНЫЕ РАБОТЫ

1. ПРОГРАММИРОВАНИЕ ЗАДАЧ ЛИНЕЙНОЙ СТРУКТУРЫ

Цель работы: приобретение практических навыков разработки и отладки простейших программ, реализующих линейные вычислительные процессы. Освоение основных принципов работы в пакете Scilab:

- работа в командном окне среды Scilab;
- основы работы в редакторе SciNotes;
- запуск приложения из редактора;
- операторы ввода/вывода в среде Scilab.

Постановка задачи: решить следующие задачи, имеющие линейную структуру алгоритма решения.

Методические указания по работе в Scilab

После запуска Scilab на экране появиться *основное окно приложения*. Окно содержит *меню*, *панель инструментов* и *рабочую область*. Также в более поздних версиях, чем версия Scilab 5.4.0, окно содержит *обозреватель фалов*, *обозреватель переменных* и *журнал команд*. Эти окна можно закрыть, если нет необходимости в их использовании, или вновь открыть через меню Инструменты в случае, если это станет необходимым.

Признаком того, что система готова к выполнению команды, является наличие знака приглашения `-->`, после которого расположен активный (мигающий) курсор. Рабочую область со знаком приглашения обычно называют *командной строкой*. Ввод команд в Scilab осуществляется с клавиатуры. Нажатие клавиши Enter заставляет систему выполнить команду и вывести результат (рис. 1.1).

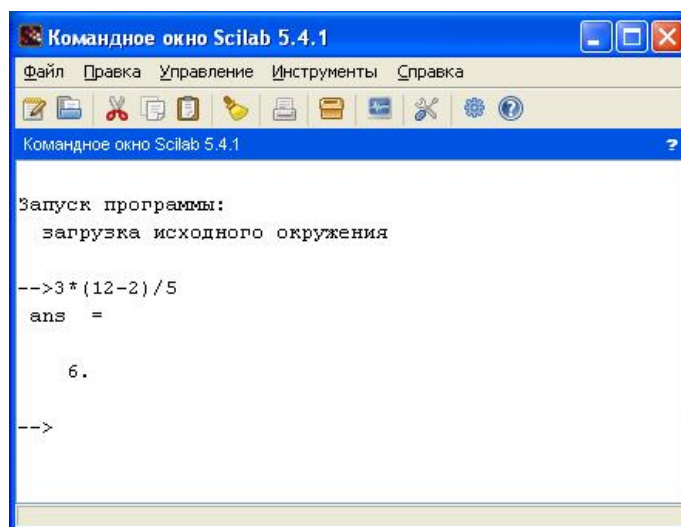


Рис. 1.1. Выполнение элементарной команды в Scilab

Вся информация в рабочей области находится или в *зоне просмотра* или в *зоне редактирования*.

Важно знать, что в *зоне просмотра* нельзя ничего исправить или ввести. Единственная допустимая операция, кроме просмотра, – это выделение информации с помощью мыши и копирование ее в буфер обмена, например, для дальнейшего помещения в командную строку.

Зона редактирования – это фактически командная строка. В ней действуют элементарные приемы редактирования: → – перемещение курсора вправо на один символ; ← – перемещение курсора влево на один символ; Home – перемещение курсора в начало строки; End – перемещение курсора в конец строки; Del – удаление символа после курсора; Backspace – удаление символа перед курсором.

Если команда заканчивается точкой с запятой «;», то результат ее действия не отображается в командной строке. В противном случае, при отсутствии знака «;», результат действия команды сразу же выводится в рабочую область (рис. 1.2).

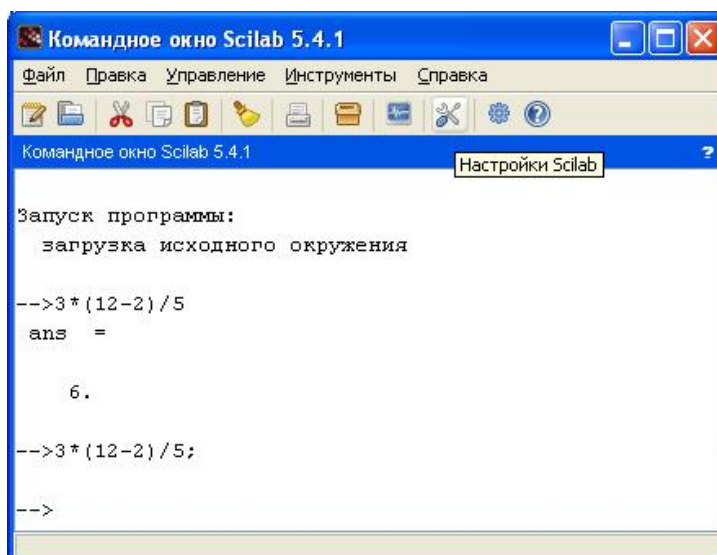


Рис. 1.2. Использование точки с запятой в Scilab

Текущий документ, отражающий работу пользователя с системой Scilab, содержащий строки ввода, вывода и сообщения об ошибках, принято называть *сессией*. Значения всех переменных, вычисленные в течение текущей сессии, сохраняются в специально зарезервированной области памяти, называемой рабочим пространством системы. При желании определения всех переменных и функций, входящих в текущую сессию можно сохранить в виде файла, саму сессию сохранить нельзя.

С помощью главного меню можно создавать, редактировать, выполнять отладку и запускать на выполнение так называемые *файлы-сценарии* Scilab.

Редактирование и отладка файлов-сценариев

Файл сценарий – это список команд Scilab, сохраненный на диске. Для подготовки редактирования и отладки файлов-сценариев служит специальный редактор SciNotes, который можно вызвать, выполнив команду главного меню Инструменты. В результате работы этой команды будет создан новый файл-сценарий. По умолчанию он имеет имя «Безымянный документ1.sce».

Ввод текста в окно редактора файла-сценария осуществляется по правилам принятым для команд Scilab. Рис. 1.3 содержит пример ввода команд для решения квадратного уравнения $3x^2 - 5x - 4 = 0$.

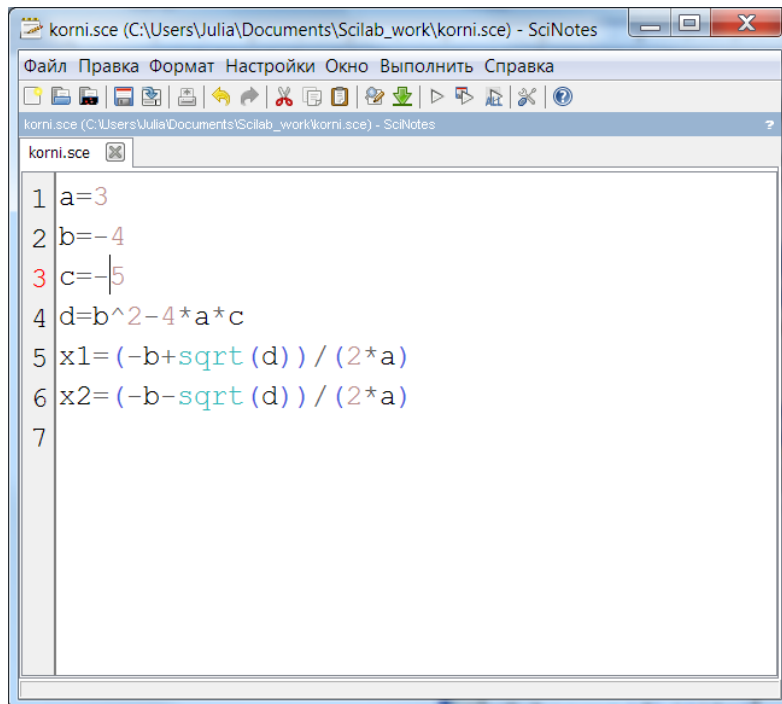


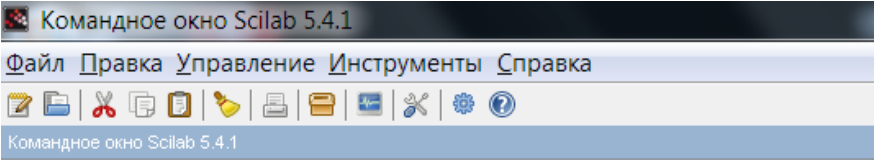
Рис. 1.3. Окно редактора файла-сценария SciNotes.
Решение квадратного уравнения

Для *сохранения* введенной информации необходимо выполнить команду «Файл → Сохранить как...» из меню редактора. Файлы-сценарии сохраняют с расширением .sce. или .sci. *Открывает* ранее созданный файл команда главного меню «Файл → Открыть».

Выполнить операторы файла-сценария можно несколькими способами:

- из меню редактора SciNotes вызвать команду «Выполнить → Сохранить и Выполнить»;
- из главного меню Scilab вызвать команду «Выполнить» и указать имя файла-сценария.

Все эти действия приведут к появлению в рабочей области результатов вычислений команд файла-сценария (рис. 1.3).



```

Командное окно Scilab 5.4.1
Файл  Правка  Управление  Инструменты  Справка

-->exec('C:\Users\Julia\Documents\Scilab_work\korni.sce', -1)

-->x1
x1 =

    2.119633

-->x2
x2 =

   - 0.7862996

-->

```

Рис. 1.4. Результаты вычислений файла сценария

Выйти из режима редактирования можно, просто закрыв окно SciNotes или выполнив команду «Файл → Выйти из SciNotes».

Основы работы в Scilab

Для выполнения простейших *арифметических операций* в Scilab применяют следующие операторы:

- + сложение;
- вычитание;
- * умножение;
- / деление слева направо;
- \ деление справа налево;
- ^ возведение в степень.

Вычислить значение *арифметического выражения* можно, если ввести его в командную строку и нажать клавишу ENTER. В рабочей области появится результат:

```

-->2.2*(4.5-2.13)+5.7/2.3-6^2
ans =
   - 28.307739

```

Для создания *текстового комментария*, т.е. строки, которая не воспринимается как команда, необходимо перед строкой поставить символы //.

Переменные в Scilab

Любая переменная до использования в формулах и выражениях должна быть определена. Для *определения переменной* необходимо набрать имя переменной, символ «=» и значение переменной. Здесь знак ра-

венства – это *оператор присваивания*, действие которого не отличается от аналогичных операторов языков программирования, то есть если в общем виде оператор присваивания записать как

имя переменной = значение выражения,

то в переменную, имя которой указано слева, будет записано значение выражения, указанного справа.

Имя переменной не должно совпадать с именами встроенных процедур, функций и встроенных переменных системы и может содержать до 24 символов. Система различает большие и малые буквы в именах переменных.

Системные переменные в Scilab

Если команда не содержит знака присваивания, то по умолчанию вычисленное значение присваивается специальной системной переменной `ans`.

Другие *системные переменные* в Scilab начинаются с символа `%`:

`%i` - мнимая единица ($\sqrt{-1}$);

`%pi` - число $\pi = 3.141592653589793$;

`%e` - число $e = 2.7182818$ (основание натурального логарифма);

`%inf` - машинный символ бесконечности (∞);

`%NaN` - неопределенный результат, «не число» («not a number», $0/0$, ∞/∞ и т.п.);

`%eps` - условный ноль `%eps = 2.220E -16`;

`%t` или `%T` – логическая единица, «истина» («true»);

`%f` или `%F` – логический ноль, «ложь» («false»);

`%z` – полином с единственным нулевым корнем и аргументом `z`;

`%s` – полином с единственным нулевым корнем и аргументом `s`.

Таблица 1.1

Элементарные математические функции

Функция	Описание функции
1	2
Тригонометрические	
<code>sin(x)</code>	Синус числа x
<code>cos(x)</code>	Косинус числа x
<code>tan(x)</code>	Тангенс числа x
<code>cotg(x)</code>	Котангенс числа x
<code>asin(x)</code>	Арксинус числа x
<code>acos(x)</code>	Арккосинус числа x
<code>atan(x)</code>	Арктангенс числа x
<code>cosh(x)</code>	Гиперболический косинус числа x
<code>sinh(x)</code>	Гиперболический синус числа x

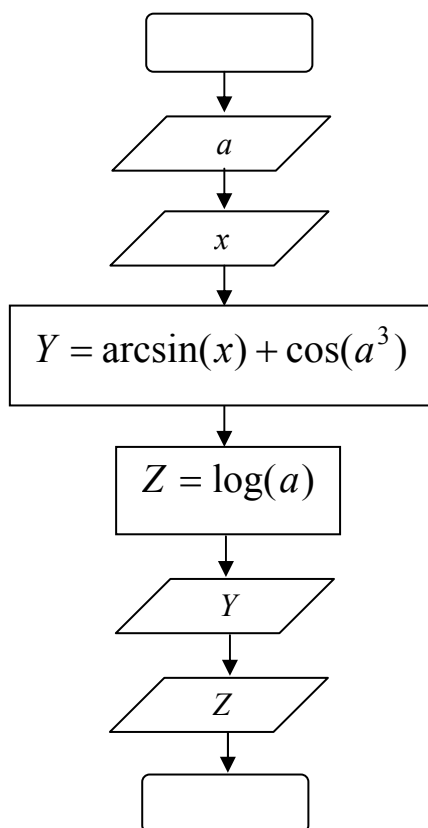
1	2
Экспоненциальные	
$\exp(x)$	Экспонента числа x
$\log(x)$	Натуральный логарифм числа x
$\sqrt{x}$	Корень квадратный из числа x
$\text{abs}(x)$	Модуль числа x
$\log_{10}(x)$	Десятичный логарифм от числа x
$\log_2(x)$	Логарифм по основанию два от числа x

Задачи

Задача 1. Составить программу для вычисления Y и Z по заданным формулам.

$$Y = \arcsin(x) + \cos(a^3), \quad Z = \log(a) \quad \text{при } a=1, \quad x=1.14.$$

Блок-схема алгоритма



Листинг программного кода

```
-->a=1;x=1.14;Y=asin(x)+cos(a^3);Z=log(a);
```

Результат работы программы

```
-->Y
```

```
Y =
    2.1110986 + 0.5231636i
```

```
-->Z
```

```
Z =
    0.
```

Задача 2. Составить программу для вычисления Y и Z (или U) по заданным формулам (по вариантам).

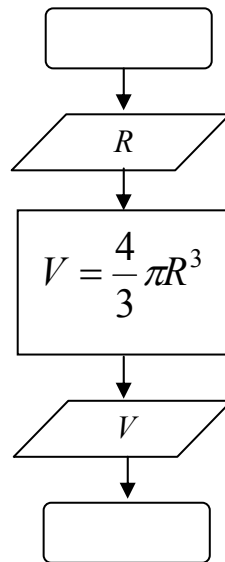
1. $Y = \arccos x + a^2, Z = \ln(x^3 + \cos a)$ при $a = 0.75; x = 0.14$	2. $Y = \lg(x/2) + a^3, Z = \operatorname{tg}(e^x + \cos a)$ при $a = 0.34; x = 0.02$
3. $Y = \sqrt{x^2 + a}, Z = \arcsin(x^3 - a)$ при $a = 0.01; x = 0.12$	4. $Y = \ln(1.5x), Z = \operatorname{arctg}(\cos x^2)$ при $a = 2.5; x = 3.11$
5. $Y = \sqrt{x^{15}} / a, Z = \cos(3.56(x + a))$ при $a = -5.1; x = 4.78$	6. $Y = \cos(x^3 + a^3), Z = e^{2x} + \operatorname{tg}(2x + a)$ при $a = 2.48; x = 0.21$
7. $Y = 2a^x + \operatorname{arctg}\left(\frac{x}{a}\right), Z = \cos x^3 + ax,$ при $a = 2.8, x = 3.29$	8. $Y = \operatorname{tg}(x - a^2) ^4, Z = e^{2x} + \operatorname{tg}(2x + a)$ при $a = 0.35; x = 0.21$
9. $Y = \ln \sin(x + a) , Z = \operatorname{tg}(xe^x)^2$ при $a = -3.4; x = 2.75$	10. $Y = \ln 2x^3 + a^{3/2}, Z = 3.7\operatorname{tg}^2 2x$ при $a = 2.53; x = 0.7$
11. $U = xe^{-yx} - \sin x + y^x + \cos^3 x,$ при $x = 0.5; y = 1.6$	12. $U = ye^x + \ln(x^4 + 1) + x^3 + y^3,$ при $x = 2.67; y = 3.1$
13. $U = x \sin^2 y - \operatorname{ctgx} + \ln^2(x + y),$ при $x = 2; y = 0.6$	14. $U = \operatorname{tg} x - x \sin^2 y + x - y ,$ при $x = 2; y = 2$
15. $U = \cos e - x^2 + y^2 - \sin x,$ при $x = -0.5; y = 0.6$	16. $U = 2^{x+y} - x \cos y + 2y - x,$ при $x = 3.4; y = -1.5$
17. $U = x + \sin y - \operatorname{tg}^2 xy + \ln x + y ,$ при $x = 0.5; y = -1.3$	18. $U = \operatorname{ctg}(x + y) - x^2 + y^2 + x - y,$ при $x = -1.5; y = 2.4$
19. $U = x^2 y - x^{-y} + \sin xy,$ при $x = 0.8; y = 0.75$	20. $U = \sin(x + y) + x - 4x^4 + \ln(x^7 + y^3) + \sin y$ при $x = 1.5; y = 2.2$
21. $U = \cos(x + y) + \ln(x^7 + y^3)^{4/3},$ при $x = 1.5; y = 2.2$	22. $U = x^3 + \sqrt{1 - x^3} + (a^2 + b^2) - e^x,$ при $x = 1.5; y = 0.2, a = 5, b = 2$

Задача 3. Вычислить объем шара V с радиусом R . Решить задачу при $R=0.2$.

Решение

Объем шара вычисляется по формуле $V = (4/3)\pi R^3$.

Блок-схема алгоритма



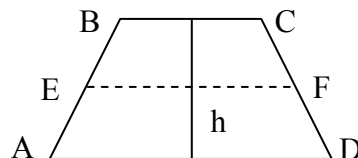
Листинг программного кода

```
-->R=0.2; V=4/3*pi*R^3
```

Результат работы программы

```
V =  
0.0335103
```

Задача 4. Вычислить площадь трапеции, представленной далее. Решить задачу при $AD=10$, $BC=6$, $h=5$.

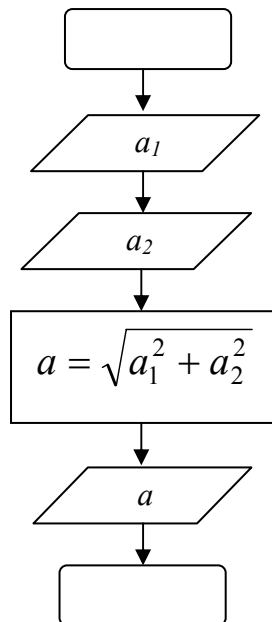


Задача 5. Вычислить длину вектора $\vec{a} = \{\vec{a}_1, \vec{a}_2\}$, где a_1 и a_2 – координаты вектора. Вычислить длину при $a_1 = 3.67$, $a_2 = 5.78$.

Решение

Длина вектора вычисляется по формуле $|\vec{a}| = \sqrt{a_1^2 + a_2^2}$.

Блок-схема алгоритма:



Листинг программного кода

```
-->a1=3.67
a1  =
    3.67
-->a2=5.78
a2  =
    5.78
-->a=sqrt(a1^2+a2^2)
```

Результат работы программы

```
a  =
    6.8466999
```

Задача 6. Найдите координаты вектора $\vec{v}$, если $\vec{v} = 2\vec{a} - 3\vec{b} - \frac{1}{2}\vec{c}$,
 $\vec{a} = \{4;1\}$, $\vec{b} = \{1;2\}$, $\vec{c} = \{4;-6\}$.

Контрольные вопросы

1. Что такое блок-схема? Для чего она нужна?
2. Как в блок-схеме обозначаются блоки ввода/вывода, начала/конца алгоритма?
3. Функция ввода в Scilab, функция вывода Scilab.
4. Для чего служит функция `printf`? Как она записывается в Scilab? Чем она отличается от функции `disp`?
5. Что означает следующая форма записи числа:
1.528E-5
Как это число выглядит в привычной форме при записи с фиксированной точкой?
6. Как в Scilab записать функции `arcsin(x)`, `log10100`, π , $\sqrt{y}$?
7. Что означают следующие записи в Scilab `x^3`; `sqrt(y)`; `abs(z)`?
8. Что означает функция `int`? Когда её удобно использовать?

2. ПРОГРАММИРОВАНИЕ ЗАДАЧ РАЗВЕТВЛЯЮЩЕЙСЯ СТРУКТУРЫ

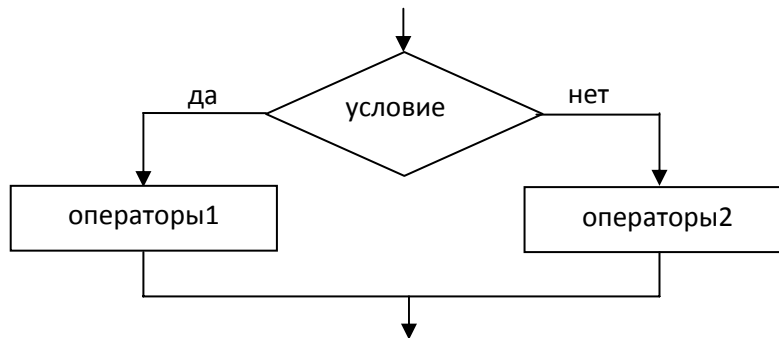
Цель работы: приобретение практических навыков разработки и отладки простейших программ, реализующих разветвляющиеся вычислительные процессы.

Постановка задачи: решить следующие задачи, имеющие разветвленную структуру алгоритма решения.

Методические указания

Условный оператор

Одним из основных операторов, реализующим ветвление в большинстве языков программирования, является условный оператор `if`. Существует обычная и расширенная формы оператора `if` в Scilab. Обычный `if` имеет вид



```
if условие
операторы1
else
операторы2
end
```

Здесь условие - логическое выражение, операторы1, операторы2 - операторы языка.

Оператор `if` работает по следующему алгоритму: если условие истинно, то выполняются операторы1, если ложно - операторы2.

В Scilab для построения логических выражений могут использоваться условные операторы: `&`, `and` (логическое и), `|`, `or` (логическое или), `~`, `not` (логическое отрицание) и операторы отношения: `<` (меньше), `>` (больше), `==` (равно), `~=`, `<>` (не равно), `<=` (меньше или равно), `>=` (больше или равно).

Расширенная форма оператора `if` имеет вид

```
if условие1
```

```

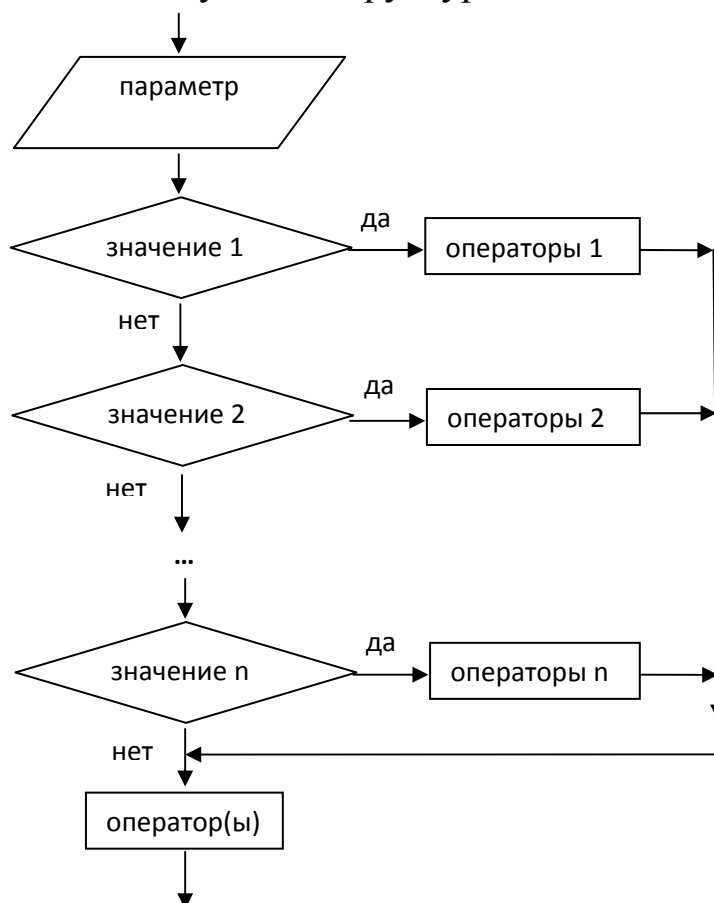
операторы1
elseif условие2
операторы2
elseif условие 3
операторы3
...
elseif условие n
операторыn
else
операторы
end

```

В этом случае оператор `if` работает следующим образом: если условие1 истинно, то выполняются операторы1, иначе проверяется условие2, если оно истинно, то выполняются операторы2, иначе проверяется условие3 и т.д. Если ни одно из условий по веткам `else` и `elseif` не выполняется, то выполняются операторы по ветке `else`.

Оператор альтернативного выбора

Еще одним способом организации разветвлений является оператор альтернативного выбора `select` следующей структуры:



```

select параметр
case значение1 then операторы1
case значение2 then операторы2
... else операторы
end

```

Оператор `select` работает следующим образом: если значение параметра равно значению1, то выполняются операторы1, иначе если параметр равен значению2, то выполняются операторы2; в противном случае, если значение параметра совпадает со значением3, то выполняются операторы3 и т.д. Если значение параметра не совпадает ни с одним из значений в группах `case`, то выполняются операторы, которые идут после служебного слова `else`.

Задачи

Задача 1. Составить программу для вычисления значения функции $U(x, y)$ при заданных значениях аргументов x и y . Вывести на экран значения функции до пятого знака после запятой.

$$u = \begin{cases} \sin(x + y) + x, & x^7 y < 0, \\ 4x^4 + \ln(x^7 + y^3), & 0 \leq x^7 y < 5, \\ \sin y, & x^7 y \geq 5. \end{cases}$$

при $x = 1.5; \quad y = 2.2$.

Примечание: Scilab по умолчанию в качестве результата выводит восемь значащих цифр. Для того чтобы контролировать количество выводимых на печать разрядов, применяют команду `printf` с заданным форматом, который соответствует правилам, принятым для этой команды в языке C.

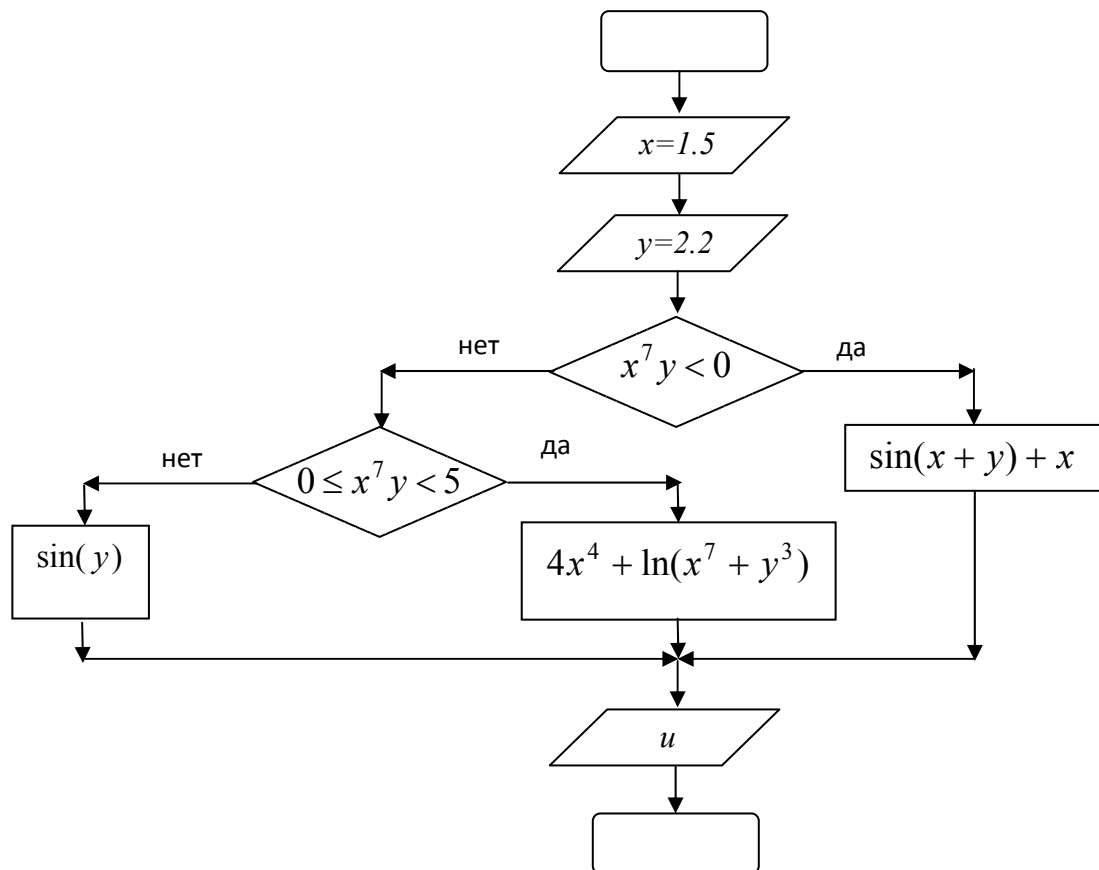
Так, для вывода числа π до четвертого знака после запятой функция `printf` будет записана следующим образом:

```

-->printf("%1.4f", %pi)
3.1415

```

Блок-схема алгоритма



Листинг программного кода

```

x=input('x= ')
y=input('y= ')
if x^7*y<0 then
u=sin(x+y)+x
elseif (x^7*y>=0) & (x^7*y<5) then
u=4*(x^4)+log(x^7+y^3)
else
u=sin(y)
end
printf("u=%1.5f",u)
  
```

Результат работы программы

```

-->exec('C:\Documents
документы\zadacha8.sce', -1)
x= 1.5
y= 2.2
u=0.80850
  
```

and

Settings\Мои

Задача 2. Составить программу для вычисления значения функции $U(x, y)$ (или $Z(x), Z(y)$) при заданных значениях аргументов x и y (x, y). Вывести на экран значения функции до третьего знака после запятой (по вариантам).

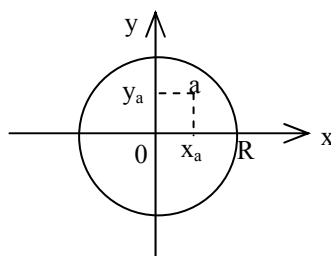
1. $U = \begin{cases} xe^{-yx}, & x + y < 0, \\ x \sin x, & 0 \leq x + y < 3, \\ y^x + \cos^3 x, & x + y \geq 3 \end{cases}$ при $x = 0.5; y = 1.6$	2. $U = \begin{cases} ye^x, & x - y < 0, \\ \ln(x^4 + 1), & 0 \leq x - y < 3, \\ x^3 + y^3, & x - y \geq 3 \end{cases}$ при $x = 2.67; y = 3.1$
3. $U = \begin{cases} x \sin^2 y, & xy \leq 1, \\ \operatorname{ctg} x, & 1 < xy < 3, \\ \ln^2(x + y), & xy \geq 3 \end{cases}$ при $x = 2; y = 0.6$	4. $U = \begin{cases} \operatorname{tg} x, & x^2 y < 0, \\ x \sin^2 y, & 0 \leq x^2 y < 5, \\ x - y , & x^2 y \geq 5 \end{cases}$ при $x = 2; y = 2$
5. $U = \begin{cases} \cos e, & xy < 0, \\ x^2 + y^2, & 0 \leq xy < 3, \\ \sin x, & xy \geq 3 \end{cases}$ при $x = -0.5; y = -1.8$	6. $U = \begin{cases} 2^{x+y}, & xy^3 < 0, \\ x \cos y, & 0 \leq xy^3 < 6, \\ 2y - x, & xy^3 \geq 6 \end{cases}$ при $x = 3.4; y = -1.5$
7. $U = \begin{cases} x + \sin y, & x - y < 0, \\ \operatorname{tg}^2 xy, & 0 \leq x - y < 3, \\ \ln x + y , & x - y \geq 3 \end{cases}$ при $x = 0.5; y = -1.3$	8. $U = \begin{cases} \operatorname{ctg}(x + y), & x^2 + y^2 < 1, \\ x^2 + y^2, & 1 \leq x^2 + y^2 < 4, \\ x - y, & x^2 + y^2 \geq 4 \end{cases}$ при $x = -1.5; y = 2.4$
9. $U = \begin{cases} x^2 y, & x + y < 0, \\ x^{-y}, & 0 \leq x + y < 4, \\ \sin xy, & x + y \geq 4 \end{cases}$ при $x = 0.8; y = 0.75$	10. $Z = \begin{cases} \frac{a}{\sqrt{x^2 + a^2}} - bx, & x > a, \\ \frac{a}{\sqrt{x^2 + a^2}} + \frac{b}{x}, & x \leq a \end{cases}$ при $x = 2n^2 + 1, b = -0.3; 0.5; 1.3; 4.6;$ $n = -1; a = 10.3$

11. $Z = \begin{cases} \sqrt{c}(x+bc), & x \geq 6, \\ \frac{\sqrt{c}}{x}(x-bc), & x < 6 \end{cases}$ при $x = 9c + b\sqrt{y}$, $y = 0.2$; $b = 0.78$; $c = 0.01; 0.02; 0.4; 0.9$	12. $Z = \begin{cases} bx + \frac{25}{\sqrt{b+x^2}}, & x < 3, \\ b\sqrt{b/x+3}, & x \geq 3 \end{cases}$ при $x = 0.3t^2$, $b = 3.5; 9.2; 4.8; 7.6$; $t = -4$
13. $Z = \begin{cases} y^2 + \frac{a+y^2}{ay}, & y \geq 1, \\ \frac{y}{3a} + \sqrt{a+3y}, & y < 1 \end{cases}$ при $y = \frac{x+a}{2x}$, $a = 0.3; 3.2; 4.7; 6.1$; $x = 0.2$	14. $Z = \begin{cases} \frac{x^2}{x+a} + \sqrt{x}, & a \geq 1, \\ \sqrt{ax} + 3x, & a < 1 \end{cases}$ при $a = \sqrt{2x^2 + 0.2c}$, $x = 1.5$; $c = 1.2; 1.5; 1.6; 6.4$
15. $Z = \begin{cases} \frac{px-5}{bx+0.5} - \sqrt{bx+5}, & bx \geq 5, \\ \frac{2p^2}{x^2+3} + \sqrt{5+2bx}, & bx < 5 \end{cases}$ при $x = 3k + 2$, $k = 8$; $p = 0.4$; $b = 1.3$	16. $Z = \begin{cases} \frac{ax^2+b}{bx+a} - \sqrt{x+ab}, & x \geq a \\ \frac{bx-a}{x^2} + \sqrt{x+a}, & x < a \end{cases}$ при $x = \frac{4.1k^2-2}{k^2}$, $k = 4$; $a = 4$; $b = 2; 1.5; 3; 10$
17. $Z = \begin{cases} \sqrt{a^2+0.2x} + \frac{b}{\sqrt{x}}, & x \geq 3, \\ \sqrt{a^2+0.2x} + \frac{b}{x}, & x < 3 \end{cases}$ при $x = 5 + 3i$, $i = 0.2$; $b = 12$; $a = 0.42; 1.5; 0.3; 1.1$	18. $Z = \begin{cases} \sqrt{x^2+a^2} + \sqrt{\frac{x}{a+0.2x}}, & x \geq 0, \\ -\sqrt{x^2+a^2} - \sqrt{\frac{x}{a+0.2x}}, & x < 0, \end{cases}$ при $x = k^3 + k + 0.1$, $k = 0.4$; $a = 2.2; 1.23; 2.8; 5.3$

19. $Z = \begin{cases} ax^2 + ax - \sqrt{\frac{b}{x+0.2}}, & x \geq 0.2, \\ \frac{ax^2 + ax}{a+x} - \sqrt{\frac{b}{x+0.2}}, & x < 0.2 \end{cases}$ при $x = \frac{2t^2 + 0.3}{2}, b = 7;$ $a = 0.5; 0.8; 1.2; 2.9$	20. $Z = \begin{cases} \frac{x-a}{\sqrt{ax}} + 2a, & x \geq 6 \\ \sqrt{ax+1} - \frac{a^2}{x}, & x < 6 \end{cases}$ при $x = 0.5t^2 - 2, t = -5;$ $a = 1.2; 7; 6.5; 14$
21. $Z = \begin{cases} ax + \sqrt{2x+b}, & x \geq 0.5, \\ \frac{\sqrt{ab}}{bx+3} - a^2x, & x < 0.5 \end{cases}$ при $b = \frac{k+0.7}{3}, x = 4, 5; 6; 1, 3; 7;$ $k = 1.4; a = 3.2$	22. $Z = \begin{cases} \frac{a}{x^2 + 1.5} \sqrt{a+x}, & x \geq 2, \\ 2\sqrt{a} - \frac{x}{a}, & x < 2 \end{cases}$ при $x = t^2 - 9a, a = 8; 3, 5; 12.3; 15;$ $t = -3$
23. $Z = \begin{cases} x^2(\sqrt{c+2} - bc), & x \geq 0, \\ \frac{b}{\sqrt{c^2+17}}, & x \leq 0 \end{cases}$ при $x = i^2 = 0.7, c = 3.6; 5.1; 11; 13.4;$ $i = -1.5; b = 2.5$	24. $Z = \begin{cases} \sqrt{a^2 - x^2} - \frac{x}{a+1}, & x > 2, \\ \frac{7x-a}{x^2} + 0.6a^2, & x \leq 2 \end{cases}$ при $x = \sqrt{k^2 + 0.6}, a = -5.7; 12; -1.9; 6.8;$ $k = -2$
25. $U = \begin{cases} x^2 - \frac{b}{\sqrt{b^2 - x}}, & x < 0.5, \\ \sqrt{x}(b + 3x^3), & x \geq 0.5 \end{cases}$ при $x = \frac{t^2}{2+t}, a = -5.7; 12; -1.9; 6.8;$ $k = -2$	26. $Z = \begin{cases} \frac{x-a}{\sqrt{a^2+1}}, & x > 5, \\ \frac{x-3}{a} + \sqrt{x^2+a^2}, & x \leq 5 \end{cases}$ при $x = 3 + 0.5t, a = 6.4; 10.5; -2.3; 11.2$ $t = 1.5$

27. $Z = \begin{cases} \sqrt{x+a}(a/x^2 - 1), & x > 0, \\ \frac{\sqrt{a} + 1 - x}{x^2 + a}, & x \leq 0 \end{cases}$ при $x = \frac{k+1.5}{k}, a = 11.3; 5.6; 4.5; 2.7;$ $t = 1.2$	28. $Z = \begin{cases} \sqrt{x^2 + a^2/4} + x^3, & x < 1.6, \\ a + \frac{x}{\sqrt{a-x}}, & x \geq 1.6 \end{cases}$ при $x = \sqrt{4+t}, a = 7.4; 2.5; 1.4; 0.5;$ $t = 1.2$
---	--

Задача 3. Определить попадает ли точка a с координатами x_a и y_a внутрь круга с радиусом R . Центр круга совпадает с началом координат.



Решить задачу при 1) $x_a = 1, y_a = 1, R = 3$; 2) $x_a = 2, y_a = 5, R = 3$.

Задача 4. Решить биквадратное уравнение $ax^4 + bx^2 + c = 0$. Решить задачу для случаев 1) $a = 4, b = -5, c = 1$; 2) $a = 3, b = 8, c = -1$; 3) $a = 1, b = 2, c = 3$.

Для решения биквадратного уравнения необходимо заменой $y = x^2$ привести его к квадратному и решить это уравнение. После этого для нахождения корней биквадратного уравнения необходимо будет извлечь корни из найденных значений y . Входными данными этой задачи являются коэффициенты биквадратного уравнения a, b, c . Выходными данными являются корни уравнения x_1, x_2, x_3, x_4 или сообщение о том, что действительных корней нет.

Алгоритм состоит из следующих этапов:

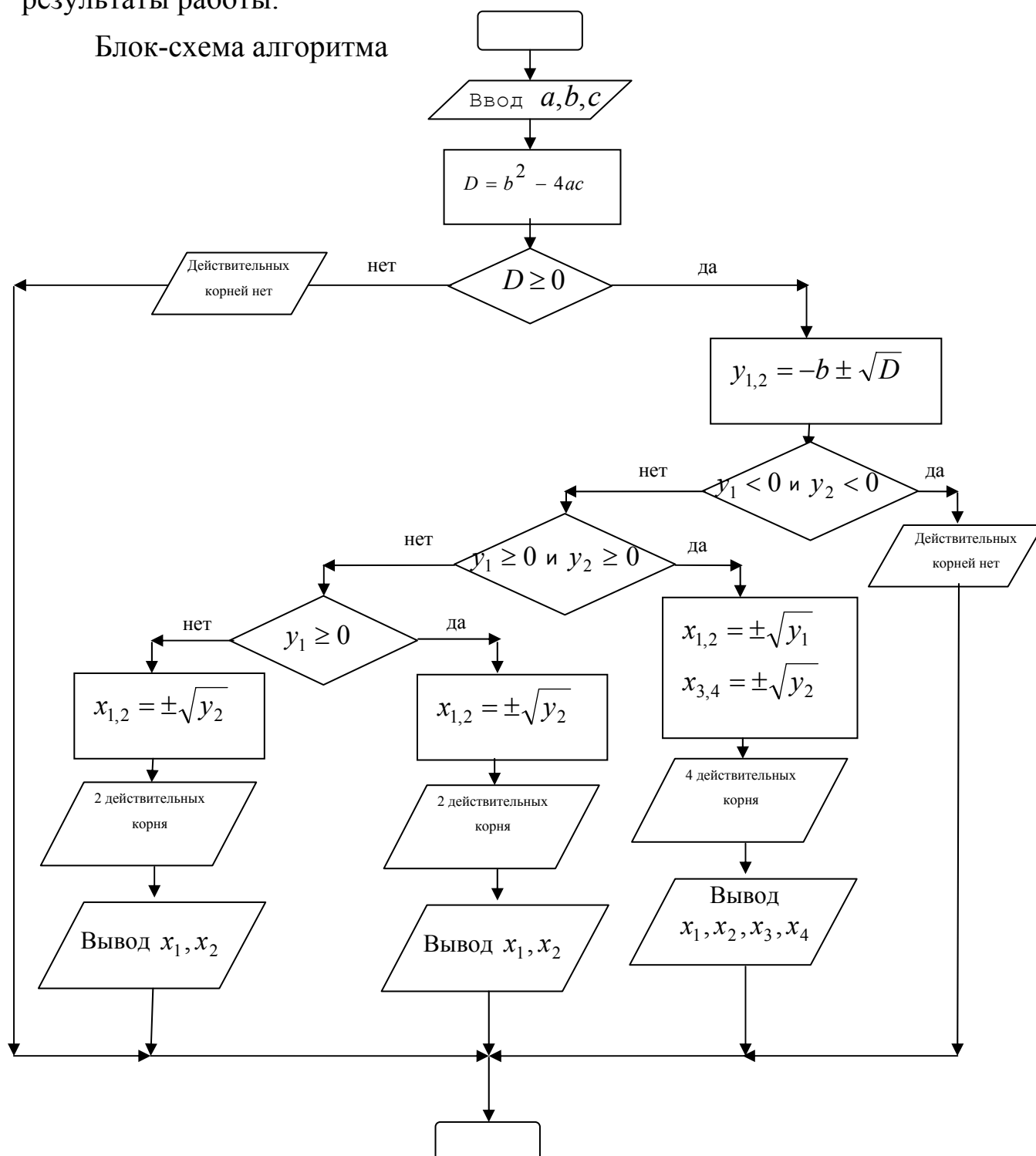
1. Ввод коэффициентов уравнения a, b, c .
2. Вычисление дискриминанта уравнения d .
3. Если $d \geq 0$, определяются y_1 и y_2 (т.е. выполняются п. 4-7), а иначе вывод сообщения «Действительных корней нет».
4. Если $y_1 < 0$ и $y_2 < 0$, то вывод сообщения «Действительных корней нет», иначе выполняется п.5.
5. Если $y_1 \geq 0$ и $y_2 \geq 0$, то вычисляются и выводятся 4 корня по формулам $x_1 = \sqrt{y_1}, x_2 = -\sqrt{y_1}, x_3 = \sqrt{y_2}, x_4 = -\sqrt{y_2}$, иначе переходим к выполнению п.6;

6. Если условия 4) и 5) не выполняются, то необходимо проверить знак y_1 .

7. Если y_1 неотрицательно, то вычисляются два действительных корня по формуле $x_{1,2} = \pm\sqrt{y_1}$ и выводится сообщение «Два действительных корня», иначе оба действительных корня вычисляются по формуле $x_{1,2} = \pm\sqrt{y_2}$ и также выводится сообщение «Два действительных корня».

Программа решения биквадратного уравнения на sci-языке приведена на листинге 9.1, далее также представлены ее вызов и результаты работы.

Блок-схема алгоритма



Листинг программного кода

```
//Ввод коэффициентов уравнения a, b и c
a=input('a=');
b=input('b=');
c=input('c=')
//Вычисление дискриминанта уравнения d
d=b^2-4*a*c;
//Если d<0, вывод сообщения «Действительных корней нет»,
иначе определяются y1 и y2
if d<0 then
disp('Действительных корней нет');
else
// Если y1<0 и y2<0, то вывод сообщения «Действительных
корней нет»
y1=(-b+sqrt(d))/(2*a);
y2=(-b-sqrt(d))/(2*a);
if (y1<0) & (y2<0) then disp('Действительных корней
нет')
//Если y1>=0 и y2>=0, то вычисляются и выводятся четыре
корня
elseif (y1>=0) & (y2>=0) then
disp('4 действительных корней');
x1=sqrt(y1);
x2=-sqrt(y1);
x3=sqrt(y2);
x4=-sqrt(y2);
disp(x1,x2,x3,x4);
//Если условия 4) и 5) не выполняются, то необходимо
проверить знак y1
//Если y1 неотрицательно, то вычисляются два корня по
формуле +-sqrt(y1),
//иначе оба корня вычисляются по формуле +- sqrt(y2).
elseif y1>=0 then
disp('2 действительных корней');
x1=sqrt(y1);
x2=-sqrt(y1);
disp(x1,x2);
else
disp('2 действительных корней');
x1=sqrt(y2);
x2=-sqrt(y2);
disp(x1,x2);
end
end
```

Результат работы программы

1)

```
-->exec('C:\Documents and Settings\Мои
документы\zadacha11.sce', -1)
a=4
b=-5
c=1
4 действительных корня
- 0.5
0.5
- 1.
1.
```

2)

```
-->exec('C:\Documents and Settings\Мои
документы\zadacha11.sce', -1)
a=3
b=8
c=-1
2 действительных корня
- 0.3458800
0.3458800
```

3)

```
-->exec('C:\Documents and Settings\Мои
документы\zadacha11.sce', -1)
a=1
b=2
c=3
```

Действительных корней нет

Если в задаче необходимо найти все корни биквадратного уравнения, а не только действительные, можно обойтись и без оператора `if`, воспользовавшись, тем, что в Scilab определены операции над комплексными числами.

Листинг программного кода

```
a=input('a=');
b=input('b=');
c=input('c=');
d=b*b-4*a*c;
x1=(-b+sqrt(d))/2/a;
x2=(-b-sqrt(d))/2/a;
y1=sqrt(x1);
y2=-y1;
y3=sqrt(x2);
y4=-y3;
disp(y1,y2,y3,y4);
```

Результат работы программы

1)

```
-->exec('C:\Documents and Settings\Мои
документы\zadacha11.sce', -1)
a=4
b=-5
c=1
- 0.5
0.5
- 1.
1.
```

2)

```
-->exec('C:\Documents and Settings\Мои
документы\zadacha11.sce', -1)
a=3
b=8
c=-1
- 1.6692213i
1.6692213i
- 0.3458800
0.3458800
```

3)

```
-->exec('C:\Documents and Settings\Мои
документы\zadacha11.sce', -1)
a=1
b=2
c=3
- 0.6050003 + 1.1687709i
0.6050003 - 1.1687709i
- 0.6050003 - 1.1687709i
0.6050003 + 1.1687709i
```

Задача 5. Решить биквадратное уравнение $ax^4 + bx^2 + c = 0$ при помощи встроенной в Scilab функции `roots`. Решить задачу для случаев 1) $a=4, b=-5, c=1$; 2) $a=3, b=8, c=-1$; 3) $a=1, b=2, c=3$. Сравнить результат с результатом в предыдущей задаче.

Примечание: Найти корни алгебраического уравнения в Scilab можно с помощью функции

`roots(p)`,

где обязательный единственный аргумент функции `p` – полином, созданный функцией `poly` и представляющий собой левую часть уравнения $P(x)=0$.

Функция `poly` описывается следующим образом:

`poly(a, "x["f1"])`

где a – число или матрица чисел, x – символьная переменная, $f1$ – необязательная символьная переменная, определяющая способ задания полинома, которая может принимать значения «roots» или «coeff» («r» или «c» соответственно). Если $f1=c$, то будет сформирован полином с коэффициентами, хранящимся в параметре a , если $f1=r$, то значения параметра a воспринимаются функцией как корни, для которых необходимо рассчитать коэффициенты соответствующего полинома. По умолчанию $f1=r$.

Например, в команде

```
->p=poly(3, 'x', 'r')
```

где 3 – это корень полинома, коэффициенты которого необходимо рассчитать. Результат запишется в переменную p :

```
p=-3+x
```

В команде

```
->p=poly(3, 'x', 'c')
```

где 3 – это коэффициент полинома. Результат запишется в переменную p :

```
p=3
```

Пример 1. Записать полином с корнями $-5, 0$ и 4

```
-->poly([-5 0 4], 'x')
```

```
ans =
```

```
      2      3
- 20x + x + x
```

Пример 2. Записать полином с коэффициентами $-5, 0$ и 4

```
-->poly([-5 0 4], 'x', 'c')
```

```
ans =
```

```
      2
- 5 + 4x
```

Пример 3. Найти корни уравнения $2x^4 - 8x^3 + 8x^2 - 1 = 0$.

Заметим, что в полиноме $2x^4 - 8x^3 + 8x^2 - 1$ отсутствует переменная x в первой степени, что означает, что соответствующий коэффициент равен 0

```
-->roots(poly([-1 0 8 -8 2], 'x', 'c'))
```

```
ans =
```

```
  2.306563
  1.5411961
- 0.3065630
  0.4588039
```

Задача 6. Решить кубическое уравнение $ax^3 + bx^2 + cx + d = 0$ по тригонометрической формуле Виета. Решить задачу для случаев 1) $a = 8, b = 7, c = -4, d = 1$; 2) $a = 1, b = -6, c = 11, d = -6$; 3) $a = 8, b = 7, c = -4, d = -1$; Сравнить результат с результатом, полученным при использовании встроенной в Scilab функции `roots`.

Примечание: Тригонометрическая формула Виета

$$q = \frac{a^2 - 3b}{9}, \quad r = \frac{2a^3 - 9ab + 27c}{54};$$

• Вычисляем

• Вычисляем $S = q^3 - r^2$.

$$\varphi = \frac{1}{3} \arccos \left(\frac{r}{\sqrt{q^3}} \right)$$

• Если $S > 0$, то вычисляем

и имеем три действительных корня:

$$x_1 = -2\sqrt{q} \cos(\varphi) - \frac{a}{3},$$

$$x_2 = -2\sqrt{q} \cos\left(\varphi + \frac{2}{3}\pi\right) - \frac{a}{3},$$

$$x_3 = -2\sqrt{q} \cos\left(\varphi - \frac{2}{3}\pi\right) - \frac{a}{3}.$$

• Если $S < 0$, то заменяем тригонометрические функции гиперболическими.

Здесь возможны следующие случаи в зависимости от знака q :

1. $q > 0$

$$\varphi = \frac{1}{3} \operatorname{Arch} \left(\frac{|r|}{\sqrt{q^3}} \right),$$

1 действительный, 2 комплексных корня.

$$x_1 = -2 \operatorname{sgn}(r) \sqrt{q} \operatorname{ch}(\varphi) - \frac{a}{3},$$

$$x_{2,3} = \operatorname{sgn}(r) \sqrt{q} \operatorname{ch}(\varphi) - \frac{a}{3} \pm i\sqrt{3} \sqrt{q} \operatorname{sh}(\varphi).$$

2. $q < 0$:

$$\varphi = \frac{1}{3} \operatorname{Arsh} \left(\frac{|r|}{\sqrt{|q|^3}} \right),$$

1 действительный, 2 комплексных корня.

$$x_1 = -2 \operatorname{sgn}(r) \sqrt{|q|} \operatorname{sh}(\varphi) - \frac{a}{3},$$

$$x_{2,3} = \operatorname{sgn}(r) \sqrt{|q|} \operatorname{sh}(\varphi) - \frac{a}{3} \pm i\sqrt{3} \sqrt{|q|} \operatorname{ch}(\varphi)$$

3. $q = 0$:

1 действительный, 2 комплексных корня

$$x_1 = -\sqrt[3]{c - \frac{a^3}{27}} - \frac{a}{3},$$

$$x_{2,3} = -\frac{a + x_1}{2} \pm \frac{i}{2} \sqrt{(a - 3x_1)(a + x_1) - 4b}.$$

• Если $S = 0$, то уравнение вырождено и имеет меньше 3 различных решений (второй корень кратности 2):

$$x_1 = 2 \operatorname{sgn}(r) \sqrt{q} - \frac{a}{3} = -2\sqrt[3]{r} - \frac{a}{3},$$

$$x_2 = \operatorname{sgn}(r) \sqrt{q} - \frac{a}{3} = \sqrt[3]{r} - \frac{a}{3}.$$

Здесь и ранее $\operatorname{sgn}(r)$ - знак r , т.е. $\operatorname{sgn}(r)=1$ при $r>0$, $\operatorname{sgn}(r)=-1$ при $r<0$, $\operatorname{sgn}(r)=0$ при $r=0$. Для определения знака числа или матрицы чисел (как вещественной, так и комплексной) существует функция $\operatorname{sign}(x)$, где x - число или матрица, знак которой надо определить. Гиперболические косинус, синус, арккосинус и арксинус описываются функциями $\cosh$, $\sinh$, acosh , asinh соответственно.

Задача 7. С помощью встроенной в Scilab функции `roots` решить уравнение. С помощью функции `poly` сделать проверку найденных корней (по вариантам).

1. $2x^3 - 8x^2 - 7x + 21 = 0$	2. $x^3 + 3x^2 + 5x + 7$
3. $5x^3 - 8x^2 - 8x + 5 = 0$	4. $3x^3 + 4x^4 + 2x = 0$
5. $2x^3 - 11x^2 + 12x + 9 = 0$	6. $x^3 - 5x^2 - 4x + 20 = 0$
7. $4x^3 - 10x^2 + 14x - 5 = 0$	8. $x^3 + 4x^2 + 5x - 1 = 0$
9. $x^3 + 0.4x^2 + 0.6x - 1 = 0$	10. $x^3 - 9x^2 + 15x + 25 = 0$
11. $x^3 - 7x^2 + 11x - 5 = 0$	12. $x^3 - 6x^2 - 6x - 2 = 0$
13. $x^3 - 3x + 2 = 0$	14. $x^3 + 2x^2 + 3x - 1 = 0$
15. $3x^3 + x^2 - 8x + 4 = 0$	16. $3x^3 + 2x^2 - 9x + 4 = 0$
17. $x^3 + 5x^2 + 8x - 4 = 0$	18. $x^3 + x + 2 = 0$
19. $3x^3 + 2x^2 - 7x = 0$	20. $x^3 - 3x^2 - 36x + 108 = 0$

Задача 8. Вывести на печать название дня недели, соответствующее заданному числу D , при условии, что в месяце 31 день и 1-е число – понедельник.

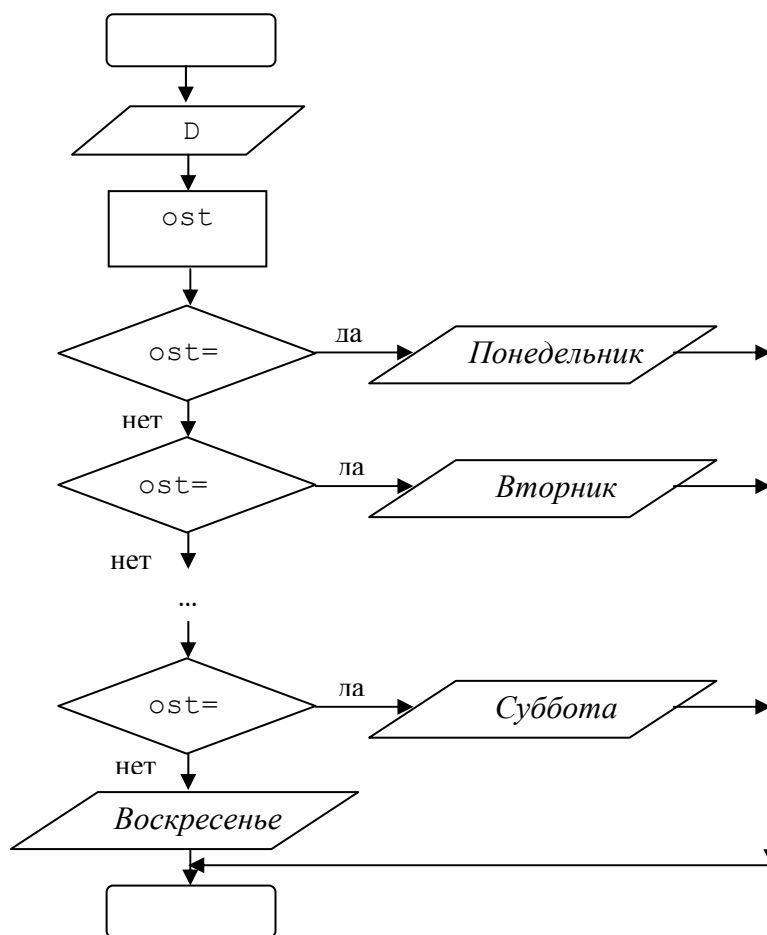
Решение

Для решения задачи воспользуемся условием, что 1-е число - понедельник. Если в результате остаток от деления заданного числа D на семь будет равен единице, то это понедельник, двойке - вторник, тройке - среда и так далее. При построении алгоритма необходимо использовать семь условных операторов. Решение задачи станет значительно проще, если при написании программы воспользоваться оператором альтернативного выбора.

Вычислить остаток от деления числа x на k можно при помощи функции `modulo(x, k)`.

Также полезной может оказаться функция `int(x/k)`, которая возвращает целую часть от деления x на k . Таким образом, остаток от деления при помощи функции `int` вычисляется по формуле $x - \text{int}(x/k) \cdot k$.

Блок-схема алгоритма



Листинг программного кода

```

D=input('Введите число от 1 до 31 ');
ost=modulo(x,7)
select ost
case 1 then disp('ПН');
case 2 then disp('ВТ');
case 3 then disp('СР');
case 4 then disp('ЧТ');
case 5 then disp('ПТ');
case 6 then disp('СБ');
else
disp('ВС');
end
  
```

Результат работы программы

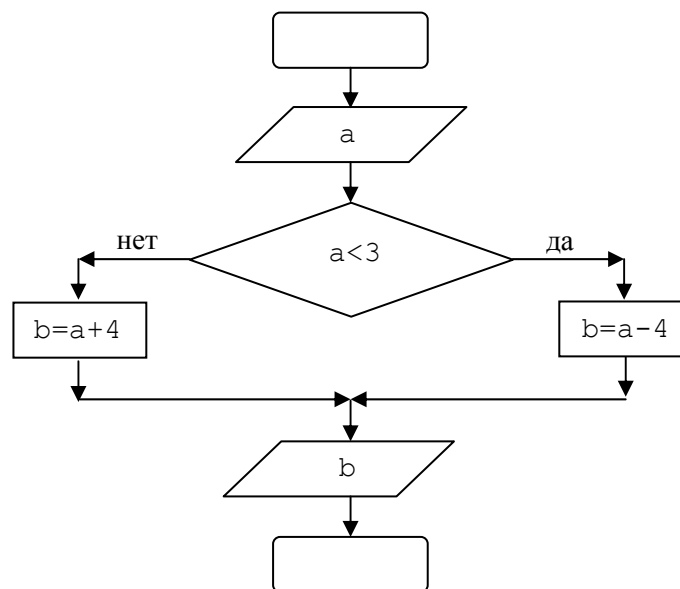
```
-->exec('C:\Documents and Settings\Мои  
документы\zadacha12.sce', -1)  
Введите число от 1 до 31 23  
BT
```

Задача 9. В старояпонском календаре был принят 12-летний цикл. Годы внутри цикла носили названия животных: крысы, коровы, тигра, зайца, дракона, змеи, лошади, овцы, обезьяны, петуха, собаки, свиньи. Написать программу, которая позволяет ввести номер года и печатает его название по старояпонскому календарю.

Контрольные вопросы

1. Как записывается и какую конструкцию имеет условный оператор в Scilab? По какому принципу он работает? Какой фигурой обозначается в блок-схеме?

2. Как следующий алгоритм записать на Scilab? Какой результат получится при $a=7$? $a=2$?



3. Что означает запись в Scilab?

```
if (a<5) & (a>3) then  
b=2*3  
end
```

4. Как найти корни алгебраического уравнения в Scilab? Для чего нужна функция `poly`?

5. Какую конструкцию в Scilab имеет оператор `select`? Как он работает?

6. В каком случае удобнее использовать оператор выбора варианта `select` вместо условного оператора `if`?

3. ПРОГРАММИРОВАНИЕ ЗАДАЧ ЦИКЛИЧЕСКОЙ СТРУКТУРЫ

Цель работы: приобретение практических навыков разработки и отладки простейших программ, реализующих циклические процессы.

Постановка задачи: решить задачи, имеющие простую циклическую структуру.

Методические указания

Циклический вычислительный процесс

Циклический вычислительный процесс, или *цикл*, – это алгоритм, в котором организовано неоднократное повторение одной и той же последовательности действий.

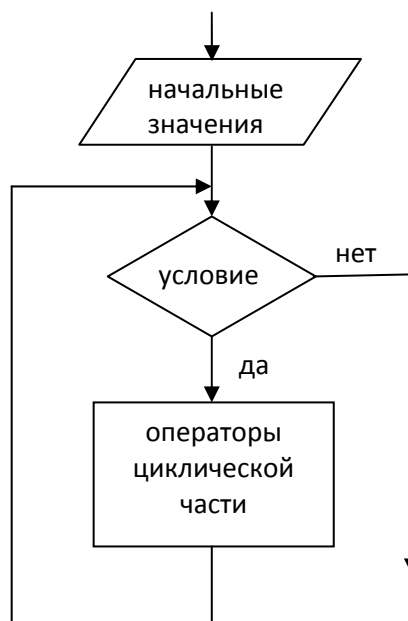
Для организации цикла необходимо предусмотреть:

1. Задание начального значения параметра (переменной) цикла.
2. Изменение переменной цикла перед каждым следующим повторением (если переменную цикла не изменять от повторения к повторению, будет организован бесконечный цикл).
3. Проверка условия окончания цикла перед каждым следующим повторением.

В Scilab для программирования циклических вычислительных процессов имеется два оператора цикла.

Оператор цикла с предусловием while

Оператор цикла while имеет вид
while условие
операторы циклической части
end



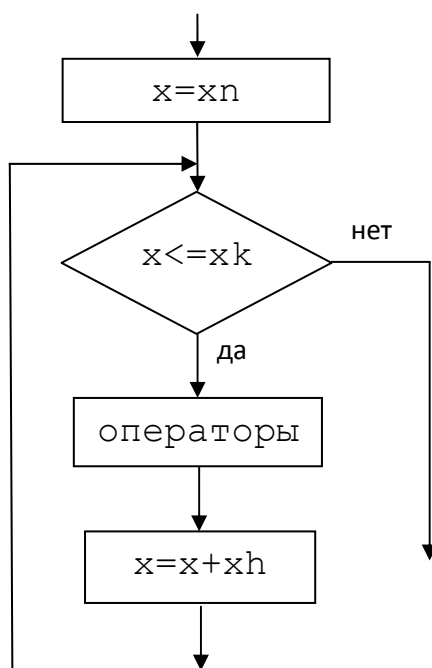
Здесь условие - логическое выражение; операторы будут выполняться циклически, пока логическое выражение истинно.

Оператор цикла `while` обладает значительной гибкостью, но не слишком удобен для организации «строгих» циклов, которые должны быть выполнены заданное число раз. В этом случае удобнее использовать оператор цикла с предусловием `for`.

Оператор цикла с предусловием `for`

Оператор цикла `for` имеет вид

```
for x=xn:hx:xk  
операторы  
end
```



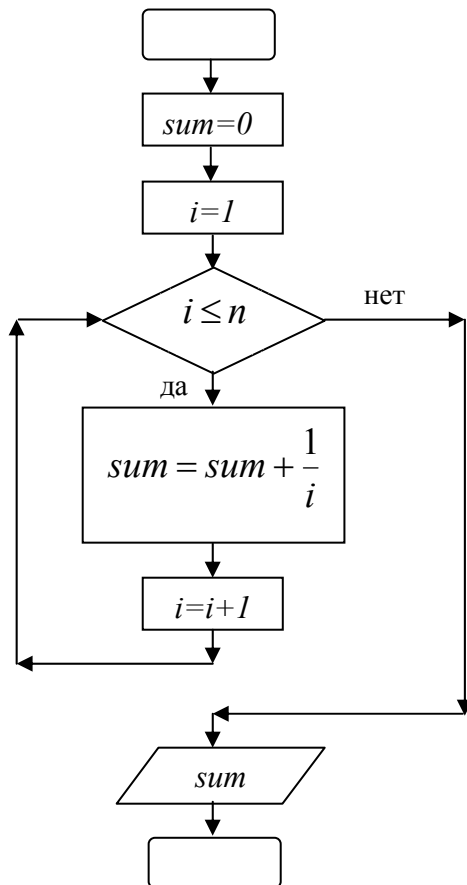
Здесь x - имя скалярной переменной - параметра цикла, x_n - начальное значение параметра цикла, x_k - конечное значение параметра цикла, h_x - шаг цикла. Если шаг цикла равен 1, то h_x можно опустить.

Выполнение цикла начинается с присвоения параметру стартового значения ($x=x_n$). Затем следует проверка, не превосходит ли параметр конечное значение ($x > x_k$). Если $x > x_k$, то цикл считается завершенным, и управление передается следующему за телом цикла оператору. Если же $x \leq x_k$, то выполняются операторы в цикле (тело цикла). Далее параметр цикла увеличивает свое значение на h_x ($x=x+h_x$). После чего снова производится проверка значения параметра цикла, и алгоритм повторяется.

Задачи

Задача 1. Найти сумму $\sum_{i=1}^n \frac{1}{i}$ двумя способами – с использованием оператора цикла с предусловием и оператора цикла с параметром.

Блок-схема алгоритма



Листинг программного кода

1) с использованием оператора цикла с параметром

```
n=input('n=')
sum=0;
for i=1:n
    sum=sum+1/i;
end
disp(sum,'sum=')
```

2) с использованием оператора цикла с предусловием

```
n=input('n=')
sum=0;
i=1;
while i<=n do
    sum=sum+1/i;
    i=i+1;
end
disp(sum,'sum=')
```

Результат работы программы

```
-->exec('C:\Documents and Settings\Мои
документы\zadacha1.sce', -1)
n=3
sum=
    1.8333333
```

Задача 2. Вычислить факториал $m!$, где m вводится с клавиатуры. Написать программу в двух вариантах: с использованием оператора цикла с предусловием и с использованием оператора цикла с параметром.

Примечание: факториал числа m — произведение всех натуральных чисел от 1 до m включительно: $m! = 1 \cdot 2 \cdot \dots \cdot m = \prod_{i=1}^m i$

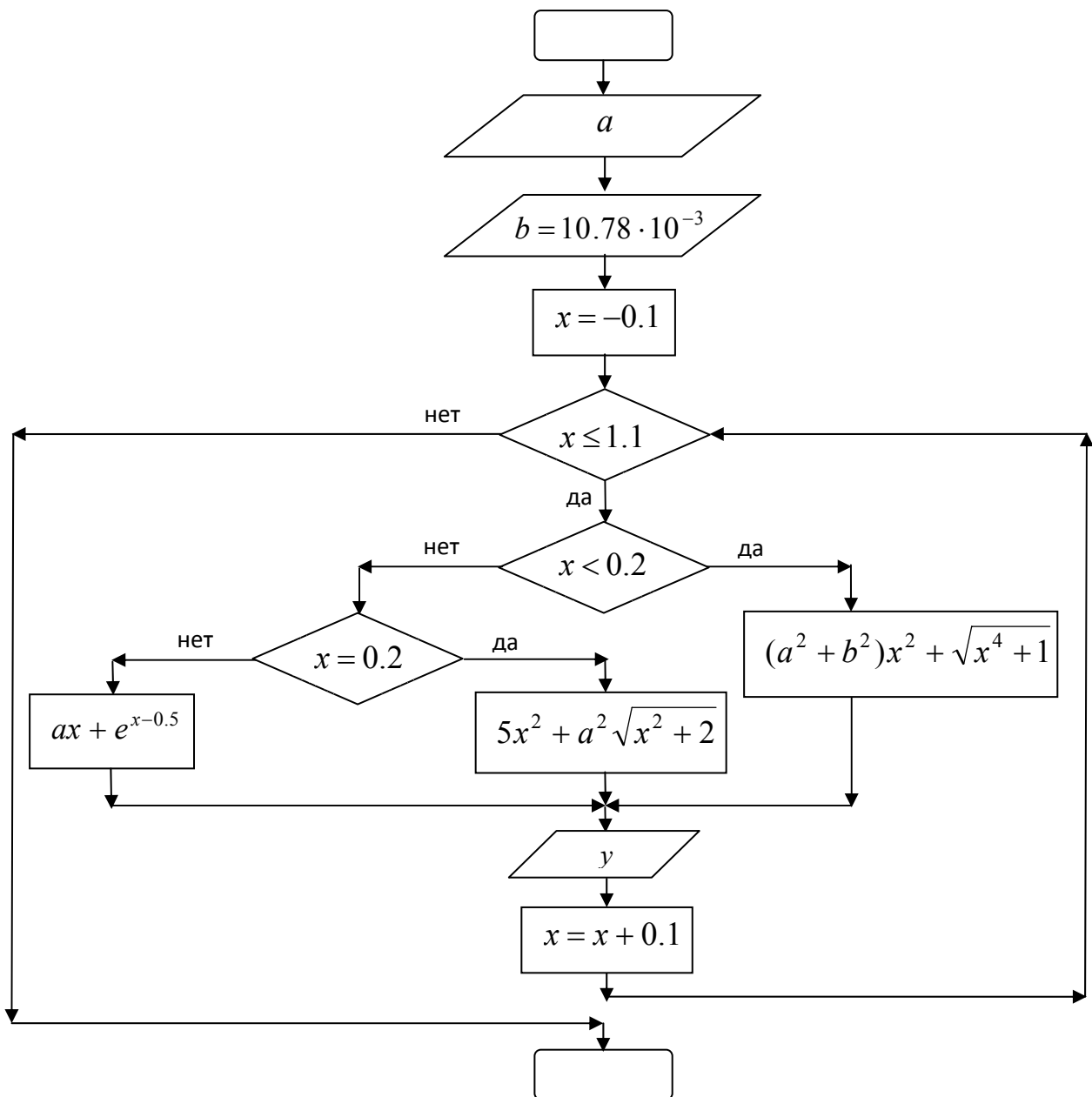
По определению полагают $0! = 1$. Факториал определён только для целых неотрицательных чисел.

Задача 3. Вычислить значения функции на заданном интервале изменения аргумента.

$$y = \begin{cases} (a^2 + b^2)x^2 + \sqrt{x^4 + 1}, & x < 0.2, \\ 5x^2 + a^2\sqrt{x^2 + 2}, & x = 0.2, \\ ax + e^{x-0.5}, & x > 0.2 \end{cases}$$

при $x \in [-0.1; 1.1]$; $\Delta x = 0.1$; $a = \{3.89; 1.3; 0.44\}$; $b = 10.78 \cdot 10^{-3}$.

Блок-схема алгоритма



Листинг программного кода

```

a=input('a=');
b=10.78*10^-3;
for x=-0.1:0.1:1.1
    if x<0.2 then
        y=(a^2+b^2)*x^2+sqrt(x^4+1);
    elseif x==0.2
        y=5*x^2+a^2*sqrt(x^2+2);
    else
        y=a*x+exp(x-0.5);
    end
    disp(y,'y=',x,'x')
end
  
```

Результат работы программы

```
-->exec('C:\Documents and Settings\Мои
документы\zadacha3.sce', -1)
a=3.89
x=      - 0.1    y=      1.1513722
x=       0.     y=       1.
x=       0.1    y=      1.1513722
x=       0.2    y=      1.5188182
x=       0.3    y=      1.9857308
x=       0.4    y=      2.4608374
x=       0.5    y=      2.945
x=       0.6    y=      3.4391709
x=       0.7    y=      3.9444028
x=       0.8    y=      4.4618588
x=       0.9    y=      4.9928247
x=       1.     y=      5.5387213
x=       1.1    y=      6.1011188
-->exec('C:\Documents and Settings\Мои
документы\zadacha3.sce', -1)
a=1.3
x=      - 0.1    y=      1.0169512
x=       0.     y=       1.
x=       0.1    y=      1.0169512
x=       0.2    y=      1.0008182
x=       0.3    y=      1.2087308
x=       0.4    y=      1.4248374
x=       0.5    y=      1.65
x=       0.6    y=      1.8851709
x=       0.7    y=      2.1314028
x=       0.8    y=      2.3898588
x=       0.9    y=      2.6618247
x=       1.     y=      2.9487213
x=       1.1    y=      3.2521188
-->exec('C:\Documents and Settings\Мои
документы\zadacha3.sce', -1)
a=0.44
x=      - 0.1    y=      1.0019872
x=       0.     y=       1.
x=       0.1    y=      1.0019872
x=       0.2    y=      0.8288182
x=       0.3    y=      0.9507308
x=       0.4    y=      1.0808374
x=       0.5    y=      1.22
x=       0.6    y=      1.3691709
x=       0.7    y=      1.5294028
x=       0.8    y=      1.7018588
x=       0.9    y=      1.8878247
x=       1.     y=      2.0887213
x=       1.1    y=      2.3061188
```

Задача 4. Вычислить значения функции на заданном интервале изменения аргумента (по вариантам).

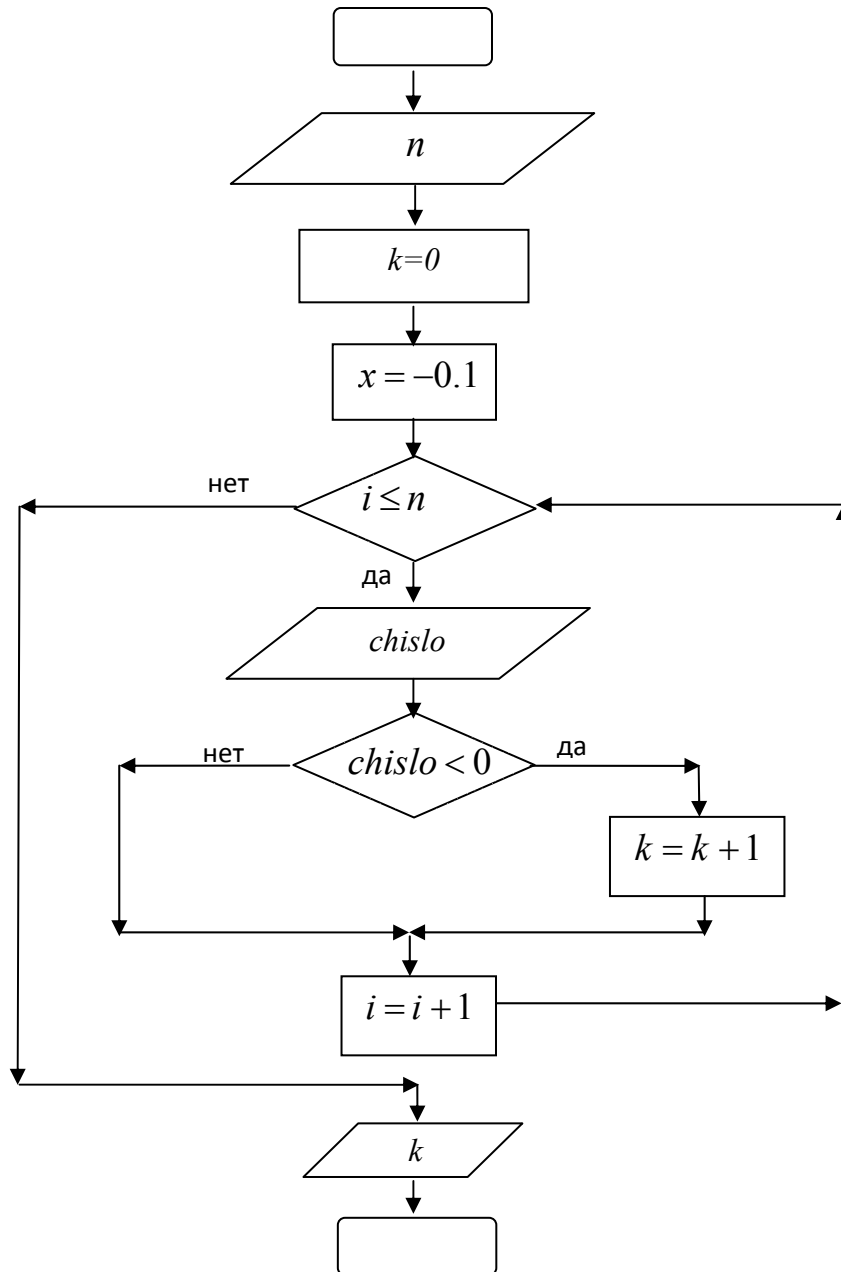
1. $y = \begin{cases} 2ax + c \sin x^2, & x < 0.5, \\ x^2 / c + b \cos x, & x = 0.5, \\ c + \sin x + \cos x, & x > 0.5 \end{cases}$ при $x \in [0.1; 0.9]$; $\Delta x = 0.1$; $a = 3.75$; $b = 0.493$; $c = \{7.31 \cdot 10^{-3}; 4.458; 10.35\}$	2. $y = \begin{cases} (a + b^2)x^3 + \sqrt{x^2 + 1}, & x < 0.5, \\ 5x^2 + a\sqrt{x^2 + 1}, & x = 0.5, \\ ax + e^{x-1}, & x > 0.5 \end{cases}$ при $x \in [-0.1; 1.2]$; $\Delta x = 0.1$; $a = \{3.899; 1.8; 0.544\}$; $b = 10.57 \cdot 10^{-3}$.
3. $y = \begin{cases} \frac{a^2 + b^3}{a + b} x^2 \sin x + \sqrt{\sin x}, & x < 0.7, \\ (a + b)\sqrt{x - e^{-x}} + x^2, & x = 0.7, \\ x^2 + \operatorname{tg} x, & x > 0.7 \end{cases}$ при $x \in [0.1; 0.9]$; $\Delta x = 0.1$; $a = \{0.2; 1.375; 6.46\}$; $b = 0.975$	4. $y = \begin{cases} a^b x + b^a x^2, & x < 1.2, \\ a + b^3 \sqrt{x^2 + 1}, & x = 1.2, \\ a^{x+b} + x, & x > 1.2 \end{cases}$ при $x \in [0.2; 1.5]$; $\Delta x = 0.2$; $b = \{3.71; 2.2; 5.55\}$; $a = 1.73$
5. $y = \begin{cases} a \sin x + b \operatorname{tg} x + (c + a) / x^2, & x < 0.6, \\ c \cos x + \sqrt{x - \sin x}, & x = 0.6, \\ \operatorname{tg} x + (a^2 + b - c) \sqrt{ \operatorname{tg} x }, & x > 0.6 \end{cases}$ при $x \in [0.1; 0.9]$; $\Delta x = 0.1$; $a = 3.71$; $b = 7.68$; $c = \{3.31 \cdot 10^{-3}; 0.66; 8.55\}$	6. $y = \begin{cases} ae^{c+x} - \ln(x^2 + 1) + x^2, & x < 0.71, \\ \sqrt{x^2 + 1} - \sin x^2 + x, & x = 0.71, \\ x + (a + b) / c, & x > 0.71 \end{cases}$ при $x \in [0.6; 1.26]$; $\Delta x = 0.11$; $c = 1.71$; $b = 1.71$; $a = \{0.31; -4.25; 7\}$
7. $y = \begin{cases} ax^3 - \operatorname{arctg} x, & x < 0.5, \\ cx^2 + b \sin x, & x = 0.5, \\ bx^2 + cx, & x > 0.5 \end{cases}$ при $x \in [0.2; 1.3]$; $\Delta x = 0.15$; $a = 0.251$; $b = -0.357$; $c = \{0.339; 1.9; 0\}$	8. $U = \begin{cases} x + \sqrt{1 - x^2} + a \sin(bx), & x < 0.8, \\ x^2 + c \cos((a + b)x), & x = 0.8, \\ \operatorname{tg} x + (a^2 + b - c) \sqrt{ \operatorname{tg} x }, & x > 0.8 \end{cases}$ при $x \in [0.1; 3]$; $\Delta x = 0.25$; $a = \{3.25; -12.8; 2.02\}$

9. $U = \begin{cases} a - \frac{2}{3 + \sin(3.6x)}, & x < 0.85, \\ 0.1x^2 - a \ln(x+1), & x = 0.85, \\ \operatorname{tg} x^2 + \frac{\operatorname{tg}^3 x}{3}, & x > 0.85 \end{cases}$ при $x \in [0.5; 1.3]; \Delta x = 0.1;$ $a = \{2.614; 0.85; 5.23\}$	10. $U = \begin{cases} 0.25x^3 + ax - 1.2502, & x < 2, \\ x + a^2 \sqrt{x} + \sqrt[3]{x+1}, & x = 2, \\ 3 \sin \sqrt{x} + 0.35 \ln a , & x > 2 \end{cases}$ при $x \in [0.1; 3]; \Delta x = 0.25;$ $a = \{3.25; -12.8; 2.02\}$
11. $U = \begin{cases} \frac{\operatorname{tg}^2 x}{5} + \frac{ax}{3}, & x < 1.5, \\ \sqrt{ 1 + 0.3x^3 } + a^x, & x = 1.5, \\ \frac{a}{2.5} + \operatorname{arctg} x^2, & x > 1.5 \end{cases}$ при $x \in [1; 3]; \Delta x = 0.25;$ $a = \{7.376; 8.33; 5.14\}$	12. $U = \begin{cases} 3x + b \ln x - 5, & x < 4.1, \\ \cos \frac{2}{x} - \frac{b}{2} \sin \frac{1}{x}, & x = 4.1, \\ x^2 - \sin^2(b - \frac{3}{x^3}), & x > 4.1 \end{cases}$ при $x \in [4; 4.2]; \Delta x = 0.02;$ $b = \{0.123; -0.93; 0.079\}$
13. $U = \begin{cases} y \sin(\ln x) - \cos(\ln x), & x < 2.5, \\ \sqrt{ 1 - 0.4x^2 } - \operatorname{arctg} \frac{x}{y}, & x = 2.5, \\ e^x - e^{-x} - 2y^2, & x > 2.5 \end{cases}$ при $x \in [2; 3]; \Delta x = 0.1;$ $y = \{7.5; -2.76; 0.157\}$	14. $U = \begin{cases} 2.75 \ln bx - \sqrt{ \ln(x+b) }, & x < 3, \\ x - 2 \sin \frac{b}{x}, & x = 3, \\ e^x + \ln x - 10x, & x > 3 \end{cases}$ при $x \in [2; 4]; \Delta x = 0.2;$ $b = \{19.01; -1.85; 0.95\}$
15. $U = \begin{cases} x^3 - 3.8ax^2, & x < 3.5, \\ 1.183ax + \cos(x^{0.52} - 2), & x = 3.5, \\ \ln^2 \left \frac{x}{a} \right + 5.62 \ln \sqrt{x}, & x > 3.5 \end{cases}$ при $x \in [2.5; 5.5]; \Delta x = 0.2;$ $a = \{2.29; -4.5; 0.733\}$	16. $U = \begin{cases} 3x - 3.87 \sin y - \ln(2+x), & x < 3.8, \\ \sqrt{x-1} - y \operatorname{tg} x, & x = 3.8, \\ 14 - e^y - e^{-2x+1.14}, & x > 3.8 \end{cases}$ при $x \in [3; 4]; \Delta x = 0.08;$ $y = \{1.11; -0.704; 2.67\}$

17. $U = \begin{cases} 0.3x - a + \sin \frac{a}{x}, & x < 1.2, \\ \cos x - e^{-\frac{x^2}{2}} + \sqrt{x+a^2}, & x = 1.2, \\ 1.08 \ln^2 x + 6a\sqrt{ x^2 - 3.52 }, & x > 1.2 \end{cases}$ при $x \in [1; 2]; \Delta x = 0.05;$ $a = \{5.09; 7.22; 1.602\}$	18. $U = \begin{cases} \frac{c}{\sin^2 x + \cos x^2}, & x < 0.9, \\ \sqrt{ \cos x + c }, & x = 0.9, \\ x^2 - \ln c + x - 3.989, & x > 0.9 \end{cases}$ при $x \in [0; 2]; \Delta x = 0.15;$ $c = \{1.307; 8.005; -4.21\}$
19. $U = \begin{cases} 2x \sin x - \frac{a}{\cos x}, & x < \frac{\pi}{2}, \\ ae^x + \sqrt{1 + e^{2x}}, & x = \frac{\pi}{2}, \\ \ln x + 2.84x - 1.83a, & x > \frac{\pi}{2} \end{cases}$ при $x \in [0; \pi]; \Delta x = \frac{\pi}{10};$ $a = \{2.026; -4.51\}$	20. $U = \begin{cases} \frac{x}{\operatorname{tg} x + b^2} + x \sin b, & x < 1.3, \\ 0.41 + \operatorname{arctg} \sqrt{x} - b, & x = 1.3, \\ \sqrt{x-1} - b \cos \sqrt{x-1}, & x > 1.3 \end{cases}$ при $x \in [0.2; 2]; \Delta x = 0.2;$ $b = \{0.547; 1.75; 2.27\}$
21. $U = \begin{cases} a \operatorname{tg} \frac{x}{a} + a \operatorname{ctg} \frac{x}{3}, & x < \frac{\pi}{3}, \\ 0.63^x - 2.34x^2 - 3.19, & x = \frac{\pi}{3}, \\ ax \operatorname{tg} x^2 + \frac{\sin(x^3 + \pi/6)}{3}, & x > \frac{\pi}{3} \end{cases}$ при $x \in \left[\frac{\pi}{24}; \frac{\pi}{2}\right] \Delta x = \frac{\pi}{24};$ $a = \{1.769; -2.9\}$	22. $U = \begin{cases} \frac{1+p}{\cos x} \ln \left \operatorname{tg} \frac{x}{2} \right + p^2, & x < 3.5, \\ \frac{1+x}{25-x^2} - 2x^{p \sin x}, & x = 3.5, \\ \left(x + \frac{p}{x} + \sin p \right) \cos x, & x > 3.5 \end{cases}$ при $x \in [2.5; 4.5]; \Delta x = 0.25;$ $p = \{1.5; -7.03; 4.8\}$

23. $U = \begin{cases} ax + e^{x \sin x}, & x < 0.44, \\ a + x^2 - \frac{1}{\sqrt{9.45 + x^2}}, & x = 0.44, \\ \frac{a \ln^2 x}{x} + x \cos a, & x > 0.44 \end{cases}$ при $x \in [0.1; 1]; \Delta x = 0.05;$ $a = \{1.12; 2.89; -0.05\}$	24. $U = \begin{cases} \frac{x \operatorname{arctg} x}{\sqrt{p^2 - 1}}, & x < 0.96, \\ e^{3x} + 1.475 p \cos x^2, & x = 0.96, \\ x^2 - \sqrt{ 4.88 p - x^3 }, & x > 0.96 \end{cases}$ при $x \in [0.5; 1.5]; \Delta x = 0.1;$ $p = \{2.302; 0.435; -0.11\}$
25. $U = \begin{cases} \frac{\ln x}{y} + \sqrt{1 - \ln^2 x}, & x < 2.7, \\ \operatorname{tg} x^2 + y \sin x, & x = 2.7, \\ 2.38/x^2 + y^2 \ln(x + 4.472)^2, & x > 2.7 \end{cases}$ при $x \in [2; 3.5]; \Delta x = 0.25;$ $p = \{1.5; -7.03; 4.8\}$	26. $U = \begin{cases} 0.495y - 2 \sin(2x) & x < \frac{2}{3}\pi, \\ \frac{\sqrt{3} \cos(x+y)/2 - y}{2(4-y)}, & x = \frac{2}{3}\pi, \\ \frac{\cos^2 y}{\sqrt{\operatorname{tg}^2(y+x) + 1}}, & x > \frac{2}{3}\pi \end{cases}$ при $x \in [0; \pi]; \Delta x = \frac{\pi}{10};$ $y = \{0.2; 1.4; -0.8\}$
27. $U = \begin{cases} \sqrt{a^2 - 4} + \sin(b^2 - 0.4), & a < 0.3, \\ 3 \ln a + \sqrt{b^4 + 1.1} , & a = 0.3, \\ 2 \sin(a + b^2) - \sqrt{a^2 + 2b^2}, & a > 0.3 \end{cases}$ при $a \in [2; 4]; \Delta a = 0.5;$ $b = \{0; 0.88; -1.45\}$	28. $U = \begin{cases} 5.495y - 6 \sin x, & x < \frac{1}{3}\pi, \\ \frac{\sqrt{2} \cos(x-y)/2 - y}{2(2+y)}, & x = \frac{1}{3}\pi, \\ \frac{\sin^2 y}{\sqrt{\operatorname{tg}^2(y+x) + 1}}, & x > \frac{1}{3}\pi \end{cases}$ при $x \in [0; \pi]; \Delta x = \frac{\pi}{10};$ $y = \{0.1; 1.2; -0.6\}$

Задача 5. Определить количество отрицательных чисел в последовательности чисел, введенной с клавиатуры.
Блок-схема алгоритма



Листинг программного кода

```
n=input('n=');  
k=0;  
for i=1:n  
    chislo=input('chislo=');  
    if chislo<0 then  
        k=k+1  
    end  
end  
disp(k, 'k=')
```

Результат работы программы

```
-->exec('C:\Documents and Settings\EmelyanovaUP\Мои  
документы\zad3_vl_cicl.sce', -1)  
n=3  
chislo=4  
chislo=-7  
chislo=-5  
k=  
    2.
```

Задача 6. Найти произведение чисел в непустой последовательности двумя способами: с использованием оператора цикла с предусловием и оператора цикла с параметром. Последовательность чисел вводится с клавиатуры.

Задача 7. Определить количество четных и нечетных чисел в последовательности чисел, введенной с клавиатуры.

Задача 8. Дано целое положительное число n . Найти наибольшее целое положительное число k , куб которого не превосходит n .

Задача 9. Найти наименьшее общее кратное (НОК) для двух чисел. Сравнить результат работы программы с результатом работы встроенной в Scilab функции `lcm`.

Задача 10. Найти наибольший общий делитель (НОД) для двух чисел. Найти наименьшее общее кратное (НОК) для двух чисел. Сравнить результат работы программы с результатом работы встроенной в Scilab функции `gcd`.

Задача 11. Определить количество цифр в числе. Число задается пользователем.

Контрольные вопросы

1. Какие операторы цикла вы знаете? Как они записываются в Scilab?

2. Чем оператор цикла с параметром отличается от оператора цикла с предусловием? От оператора цикла с постусловием? В каком случае его удобней использовать?

3. Нарисуйте блок-схему циклического процесса. Какие основные блоки должны в ней содержаться?

4. Дана запись в Scilab

```
for i=2:7  
s=i+2  
end
```

С каким шагом изменяется параметр i ? Сколько итераций пройдет алгоритм? Чему будет равна s в результате работы этого цикла?

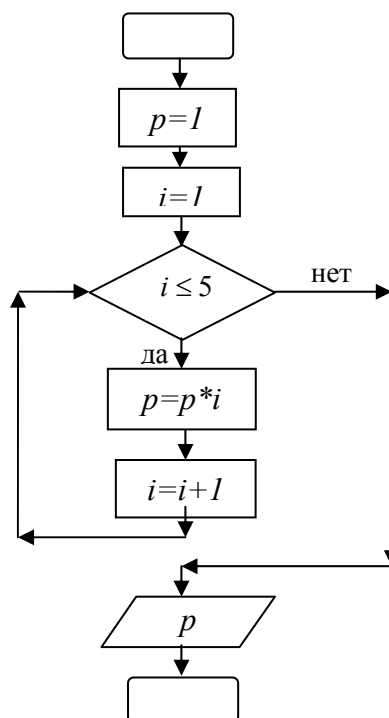
5. Как в Scilab записать следующий алгоритм?

```
a=5  
пока a>3 выполнять  
x=2*a  
a=a-1  
конец
```

Чему будет равно x в результате работы этого алгоритма?

6. Как записать в Scilab функции e^x ; $\arctg(x)$?

7. Как в Scilab реализовать представленный ниже алгоритм? Как изменится конечный результат, если начальное значение переменной p задать равным 0?



4. ПРОГРАММИРОВАНИЕ ЗАДАЧ СЛОЖНОЙ ЦИКЛИЧЕСКОЙ СТРУКТУРЫ

Цель работы: приобретение практических навыков разработки и отладки программ, реализующих сложные циклические процессы.

Постановка задачи: решить задачи, имеющие сложную циклическую структуру (два цикла и более).

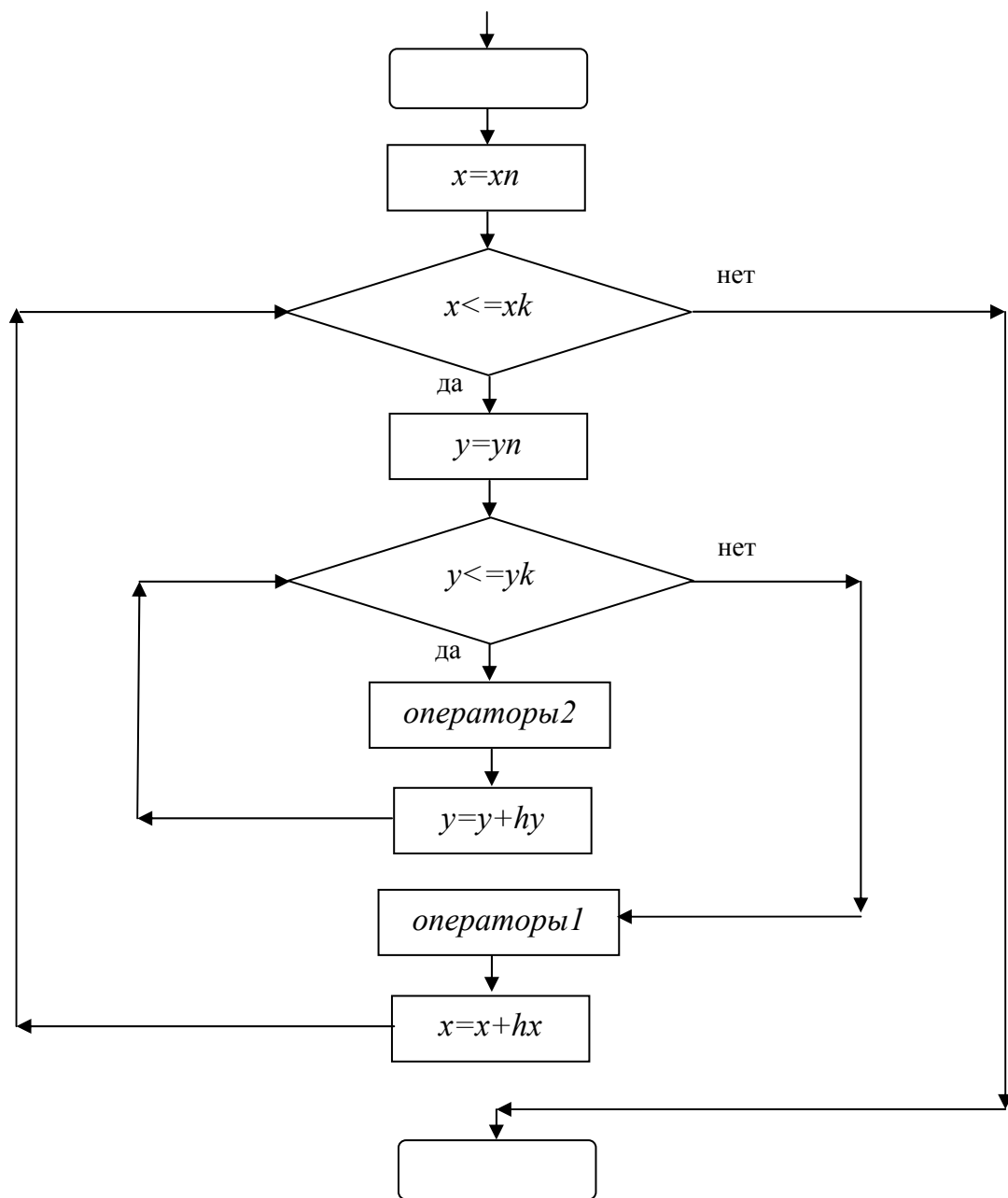
Методические указания

Вложенные циклы

Циклические вычислительные процессы вложенной структуры характеризуются наличием более, чем одного цикла и при этом внутренние циклы также выполняются циклически, т. е. организован принцип «цикл в цикле».

Далее представлены общий вид (два способа) и блок-схема циклического вычислительного процесса вложенной структуры с двумя циклами.

for x=xn:hx:xk	x=xn
	while x<=xk
for y=yn:hy:yk	y=yn
	while y<=yk
операторы2	операторы2
	y=y+yk
end	end
операторы1	операторы1
	x=x+xk
end	end



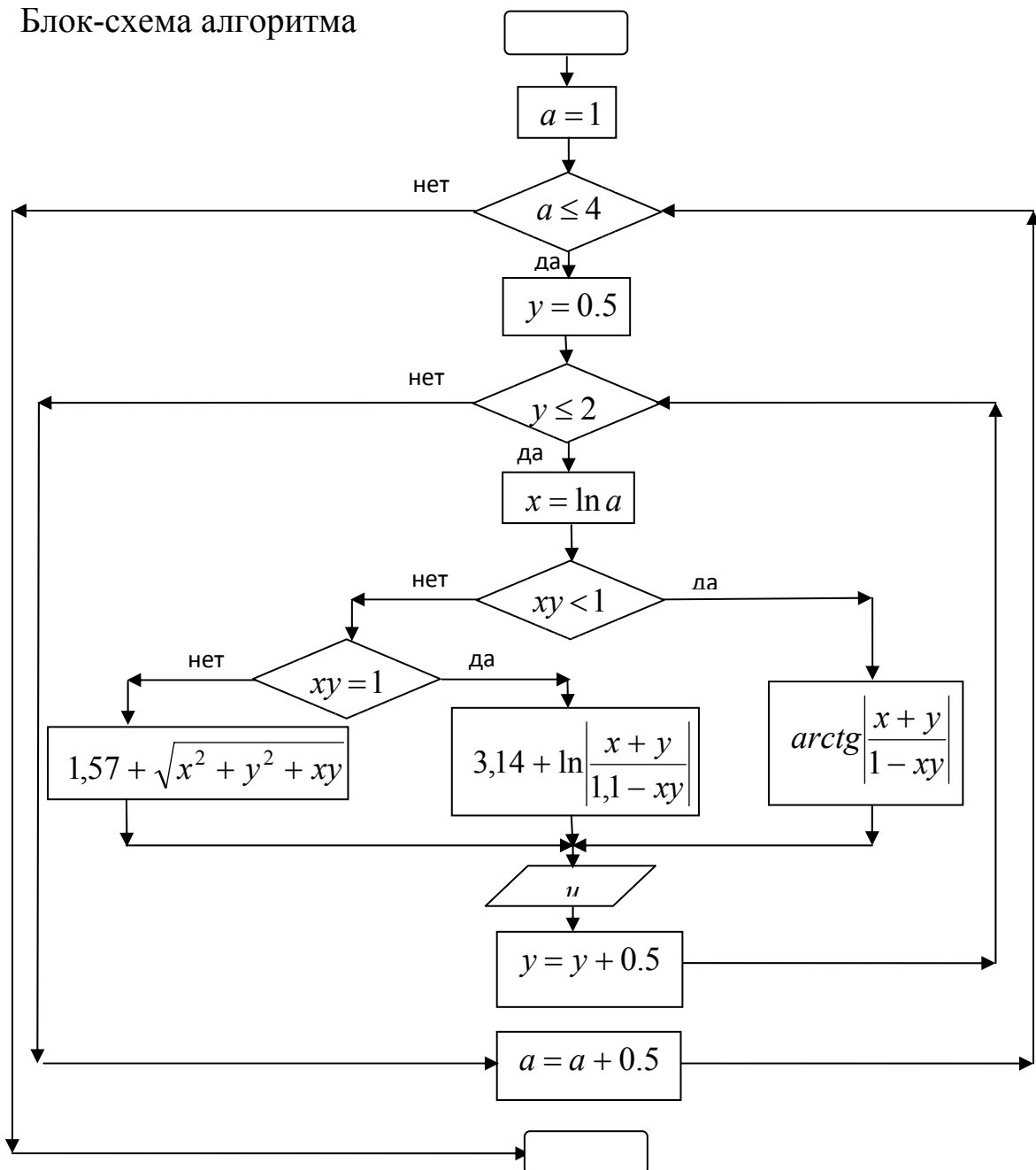
Задачи

Задача 1. Вычислить значения функций для всех сочетаний значений аргументов

$$U = \begin{cases} \operatorname{arctg} \left| \frac{x+y}{1-xy} \right|, & xy < 1, \\ 3,14 + \ln \left| \frac{x+y}{1,1-xy} \right|, & xy = 1, \\ 1,57 + \sqrt{x^2 + y^2 + xy}, & xy > 1. \end{cases}$$

при $x = \ln a$; $a = [1;4]$; $\Delta a = 0,5$; $y = [0,5;2]$; $\Delta y = 0,5$.

Блок-схема алгоритма



Листинг программного кода

```

a=1;
while (a<=4) do
    y=0.5
    while (y<=2) do
        x=log(a)
        if x*y<1 then
            u=atan(abs((x+y)/(1-x*y)))
        elseif x*y==1 then
            u=3.14+log(abs((x+y)/(1.1-x*y)))
        else
            u=1.57+sqrt(x^2+y^2+x*y)
        end
        disp(u,'u= ')
        y=y+0.5
    end;
    a=a+0.5
end

```

Результат работы программы

```

-->exec('C:\Documents
документы\zadacha7.sce', -1)

```

				and	Settings\Мои
a=	1.	y=	0.5	u=	0.4636476
a=	1.	y=	1.	u=	0.7853982
a=	1.	y=	1.5	u=	0.9827937
a=	1.	y=	2.	u=	1.1071487
a=	1.5	y=	0.5	u=	0.8488564
a=	1.5	y=	1.	u=	1.1706069
a=	1.5	y=	1.5	u=	1.3680025
a=	1.5	y=	2.	u=	1.4923575
a=	2.	y=	0.5	u=	1.0697595
a=	2.	y=	1.	u=	1.3915101
a=	2.	y=	1.5	u=	3.5116935
a=	2.	y=	2.	u=	3.9921369
a=	2.5	y=	0.5	u=	1.2053906
a=	2.5	y=	1.	u=	1.5271411
a=	2.5	y=	1.5	u=	3.6828239
a=	2.5	y=	2.	u=	4.1530544
a=	3.	y=	0.5	u=	1.2960005
a=	3.	y=	1.	u=	3.3881203
a=	3.	y=	1.5	u=	3.8293954
a=	3.	y=	2.	u=	4.2910611
a=	3.5	y=	0.5	u=	1.3607798
a=	3.5	y=	1.	u=	3.5250391
a=	3.5	y=	1.5	u=	3.9571656
a=	3.5	y=	2.	u=	4.4116441
a=	4.0	y=	0.5	u=	1.409534
a=	4.0	y=	1.	u=	3.6455978
a=	4.0	y=	1.5	u=	4.0702507
a=	4.0	y=	2.	u=	4.5186269

Задача 2. Вычислить значения функций для всех сочетаний значений аргументов (по вариантам).

1. $U = \begin{cases} a + \frac{2}{b} + 4 \operatorname{arctg} x, & 1 \leq a \leq b, \\ 2.549 + \ln \left x + b^2 - \frac{1}{a} \right , & \text{в ост. сл.} \end{cases}$ при $a = 2.1 \ln x + 1; x = [3; 7];$ $\Delta x = 1; b = [1.5; 4]; \Delta b = 0.5$	2. $U = \begin{cases} 16.5x + 2b^2 - 1.25x^3, & 1 \leq x < 9, \\ b^2 + 4.5, & b < x < 1, \\ b - x + \operatorname{tg}(ax), & \text{в ост. сл.} \end{cases}$ при $x = 7.65 \sin a + 0.24e^{8.76a};$ $a = [0; 1]; \Delta a = 0.25; b = [0.5; 0.8];$ $\Delta b = 0.1$
3. $U = \begin{cases} y + \frac{x-b}{ y-a }(a-b), & x < b \text{ или } y < a, \\ (y+x)^2 + \sqrt{a^2 + b}, & \text{в ост. сл.} \end{cases}$ при $x = a \sin \alpha; y = b \cos \alpha;$ $a = 4.5; b = [3; 6]; \Delta b = 1;$ $\alpha = [0; \pi]; \Delta \alpha = \frac{\pi}{6}$	4. $U = \begin{cases} \sqrt{2y + 6.7} + e^a, & x - 2y \geq 0 \text{ или } ax \leq 0, \\ \operatorname{tg}\left(x^2 + \frac{xy}{8}\right) + \ln(ax), & \text{в ост. сл.} \end{cases}$ при $y = 3 \sin x + 2.5; a = [-2; 1];$ $\Delta a = 0.5; x = [0.1; 0.7]; \Delta x = 0.2$
5. $z = 2U\sqrt{x^2 + 4a^2}$ при $U = \begin{cases} a(x^2 + 1 + \sin x)^2, & x \leq 0.5, \\ x^2 + ax + b, & x > 0.5; \end{cases}$ $b = 0.39; a = [0.51; 0.53];$ $\Delta a = 0.01; x = [0.1; 1.2];$ $\Delta x = 0.2$	6. $U = \begin{cases} (x^2 + a^2) \sin x, & x < 0.7, \\ x^3 + a \sin x, & x = 0.7, \\ \frac{1.49 \cos x}{12.3 + a/0.8} x, & x > 0.7 \end{cases}$ при $a = 0.5; x = [0.1; 0.9];$ $\Delta x = 0.1; b = [1.6; 2.3];$ $\Delta b = 0.3$
7. $U = \begin{cases} ax + b(x^3 - 1), & x < 0.3, \\ (a+b)^2(x + \sin x), & x = 0.3, \\ x + 2ba(x^2 - 1), & x > 0.3 \end{cases}$ при $a = 0.5; x = [0.1; 0.9];$ $\Delta x = 0.1; b = [1.6; 2.3];$ $\Delta b = 0.3$	8. $U = \begin{cases} bn + a^2(x+1), & x < 2.3, \\ nx + (b-x)^2, & x > 2.3 \end{cases}$ при $n = ax^5 + \frac{x^3}{2a}; x = a^2 + 2a - 1;$ $a = [0.4; 1.9]; \Delta a = 0.4;$ $b = [0.5; 1.2]; \Delta b = 0.1$

9. $U = \begin{cases} (\sin x + (ax + c))^2, & x < 1, \\ \cos x + ax + c^2, & x = 1, \\ (ax + c)^3, & x > 1 \end{cases}$ при $x = [0.5; 1.2]; \Delta x = 0.2;$ $a = [1.2; 1.7]; \Delta a = 0.15;$ $c = -1.2$	10. $U = \begin{cases} (\sin x + \operatorname{tg} x)^2 \frac{a}{b}, & x < 0.4, \\ ax^2 + b^2, & x \geq 0.4 \end{cases}$ при $x = a^4 + 5.2 \sin(ab);$ $b = [0.4; 1.2]; \Delta b = 0.15;$ $a = [1.5; 2.1]; \Delta a = 0.15$
11. $U = \begin{cases} \frac{\sqrt{l^2 + a^2 + b^2}}{\cos(l + a + b)}, & -1 < l < a - b, \\ \frac{\ln \operatorname{tg} l + \operatorname{tg}^2 a + b }{4\sqrt{5}}, & \text{в ост. сл.} \end{cases}$ при $l = \operatorname{tg}^2 \frac{a}{2} + \ln \sqrt{\frac{b^2 + a^2}{4}};$ $a = [0; 1]; \Delta a = 0.25;$ $b = [0.2; 1]; \Delta b = 0.2$	12. $U = \begin{cases} \frac{2k+1}{4} \sqrt{a^2 + ab + k}, & 2 < k + ab < a^2, \\ \ln 2k - 2.5 \sin(\operatorname{arctg}(ak)) , & \text{в ост. сл.} \end{cases}$ при $k = \frac{2}{a} + b^2 - \sqrt{a + b};$ $a = [0.5; 1.5]; \Delta a = 0.3;$ $b = [0.5; 1.5]; \Delta b = 0.25$
13. $U = \begin{cases} (a^2 + 1)(\sin x + \operatorname{tg} x), & 0.1 < x < 0.5, \\ (a^2 + 1)^3 x, & x \leq 0.1, \\ x(\sin x + \ln x)^3, & x > 0.5 \end{cases}$ при $a = [0.31; 0.39]; \Delta a = 0.02;$ $\Delta x = [0; 0.7]; \Delta x = 0.1$	14. $z = \left(\frac{a}{a + U} - U^2 \right) \cos^3 x \quad \text{при}$ $U = \begin{cases} \ln \left \frac{a}{x} \right + \sqrt{\sin^2 a}, & 0.5 < a + x < 1, \\ e^{ax} - \sqrt{ 2 + \cos x \sin a }, & \text{в ост. сл.} \end{cases}$ при $x = [0.1; 0.9]; \Delta x = 0.2;$ $a = [0.2; 1]; \Delta a = 0.2$
15. $U = \begin{cases} \sqrt{2 - \cos^3(l + b)}, & 0.7 \leq l \leq 1, \\ a^2 \sin^2(bl) + \operatorname{tg} a, & \text{в ост. сл.} \end{cases}$ при $l = e^{a \sin \frac{b}{a}} \sin a; a = [0.3; 0.8];$ $\Delta a = 0/1; b = [0; 1]; \Delta b = 0.25$	16. $U = \begin{cases} 2y^{c \sin d} + \cos y, & 0.3 < y < 1, \\ \frac{y + c}{\sin c + d^2}, & \text{в ост. сл.} \end{cases}$ при $y = c + e^{-d^2}; c = [0; 1];$ $\Delta c = 0.25; d = [0.1; 0.5]; \Delta d = 0.2$

17. $U = \begin{cases} 0.93e^{-ap} + \sin b, \cos(ab) < p < 1, \\ \ln \cos(2p) - \sin(ab) , & \text{в ост. сл.} \end{cases}$ при $p = 0.385 \ln a + 2.81 \sin b$; $a = [0.5; 1.5]; \Delta a = 0.5$; $b = [0.5; 1]; \Delta b = 0.1$	18. $U = \begin{cases} \ln e^l + 2 \sin p , & 0.5 < l < 5, \\ \operatorname{arctg}(l + q) + \ln p^2, & \text{в ост. сл.} \end{cases}$ при $l = \sqrt{\operatorname{tg}^2 p + 4 \operatorname{tg} q}$; $p = [0.3; 1.1]$; $\Delta p = 0.2$; $q = [0; 1.5]; \Delta q = 0.5$
19. $U = \begin{cases} h - 3e^{4d}, & hd < 0.5, \\ h^3(1 + 4de^h), & hd \leq 0.5 \end{cases}$ при $d = \frac{7.93n_1}{n_1 + e^{1.42/n_2}} + \frac{2.2138}{12.93 + n_1n_2}$; $h = \frac{\sqrt{n_1} \sin n_2}{n_1 + e^{-n_2}}$; $n_1 = [1; 5]; \Delta n_1 = 1$; $n_2 = [7; 9]; \Delta n_2 = 0.5$	20. $U = \begin{cases} y - \frac{2}{z} - 4.13, & y \geq z, \\ (y + z)^2, & y < z \end{cases}$ при $y = \sin^2 a + 3.64x^2(\ln x - 1)$; $z = \frac{\cos x - \sqrt{x^2 + 1.4a}}{3.06a - 2.5}$; $a = [0.5; 1.5]; \Delta a = 0.25$; $x = [-0.4; -0.1]; \Delta x = 0.1$
21. $U = \begin{cases} 10.6k^2 + 8l - 1.33kl, & k < lt , \\ 0.71 + l^2, & k \geq lt \end{cases}$ при $k = \frac{3.96\sqrt{ b + t^2}}{\sqrt{1.5 + b^2}}$; $l = \cos \operatorname{arctg} \frac{b}{t} + \cos^2(1.5 + b)$; $b = [0; 0.8]; \Delta b = 0.2$; $t = [1.5; 2]; \Delta t = 0.25$	22. $U = \begin{cases} 1.38d^3 + g^2, & d < g + k_1, \\ d^2 + 20g, & d < g + k_1 \end{cases}$ при $d = 2k_1 + \sin \sqrt{k_2^3} + \frac{0.22k_1^3}{e^{2k_1 + k_2}}$; $g = \frac{0.2}{\cos^3(3.14 - k_2)} - \sqrt{k_1^2 + 4k_2^2}$; $k_1 = [-0.5; 0.5]; \Delta k_1 = 0.2$; $k_2 = [0.2; 1.05]; \Delta k_2 = 0.25$
23. $U = \begin{cases} \frac{f+l}{1.64f}, & f+l \leq x+y , \\ f^2 + 10l + 1, & f+l > x+y \end{cases}$ при $f = x^3 - \frac{y - 3.21}{8.53}$; $l = \frac{x - y}{3x + 4y}$; $x = [0; 1]; \Delta x = 0.2$; $y = [2; 4]; \Delta y = 0.5$	24. $U = \begin{cases} e^{3(f+2g)} + y, & f \leq g, \\ \frac{fg + 2.71}{8.19}, & f > g \end{cases}$ при $f = \sin^2 x + 3.47 \frac{x^2 + y}{y^2 - x}$; $g = \frac{\operatorname{tg} x + y^2 \ln x }{a}$; $x = [0.5; 0.9]; \Delta x = 0.1$; $y = [-3; 3]; \Delta y = 2$; $a = 2.88$

25. $U = \begin{cases} \sqrt{ p } + \sqrt{ q }, & p \geq q , \\ \sqrt{ p-q }, & p < q \end{cases}$ при $p = (a + 4b + \sin a) \cos b$; $q = \sin^4(a + b) + \sqrt{\operatorname{tg}^3 a}$; $a = [0.4; 1.3]; \Delta a = 0.2$; $b = [-0.5; 0.5]; \Delta b = 0.25$	26. $U = \begin{cases} xy + \frac{2}{xy}, & xy \geq 2.2, \\ x^2 + 4d^2 + 3xy, & x < 2.2 \end{cases}$ при $x = 1.82e^{\sqrt{c+1}} + 0.96d^2$; $y = \frac{\sin^2 c + 1}{\cos d^2 + 1.21} + a$; $c = [0.5; 2.5]; \Delta c = 0.5$; $d = [-0.1; 0.9]; \Delta d = 0.2$
27. $U = \begin{cases} \operatorname{arctg} \left \frac{x+y}{1-xy} \right , & xy < 1.1, \\ 8.24 + \ln \left \frac{x-y}{2-xy} \right , & xy = 1.1, \\ 10.45 + \sqrt{x^2 + y^2 + xy}, & xy > 1.1 \end{cases}$ при $x = \ln a$; $a = [1; 6]; \Delta a = 0.5$; $y = [0.5; 2]; \Delta y = 0.2$	28. $U = \begin{cases} 8a + \frac{7}{b} + 4\operatorname{arctg} x, & 1 \leq a \leq 0.5b, \\ 4.67 + \ln \left 7x + 2b^3 + \frac{1}{a} \right , & \text{в ост. сл.} \end{cases}$ при $a = 2.1 \ln x + 1$; $x = [3; 7]$; $\Delta x = 0.5$; $b = [1.5; 4]; \Delta b = 0.2$

Контрольные вопросы

1. Нарисуйте блок-схему циклического процесса с двумя вложенными циклами.

2. Дана запись в Scilab:

```
for i=1:3
for i=1:4
s=i+j
end
end
```

Сколько итераций пройдет алгоритм? Чему будет равна s в результате работы этого цикла?

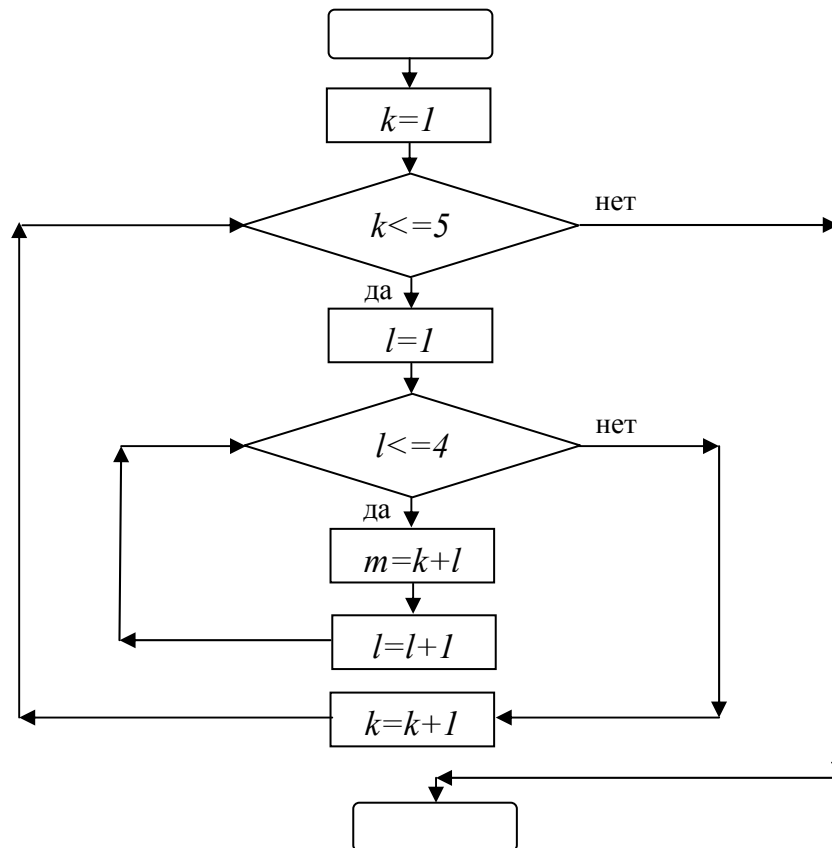
3. Как в Scilab записать следующий алгоритм?

```
a=5
b=0
пока a>0 выполнять
пока b<5 выполнять
```

$x=b+a$
 $a=a-1$
 $b=b+1$
конец
конец

Чему будет равно x в результате работы этого алгоритма?

4. Как в Scilab реализовать представленный далее алгоритм?



5. ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ ОДНОМЕРНЫХ МАССИВОВ. ФУНКЦИИ

Цель работы: приобретение практических навыков работы с одномерными массивами. Приобретение навыков работы в написании и отладке программ с использованием функций.

Постановка задачи: решить задачи с использованием одномерных массивов.

Методические указания

Массивы и матрицы в Scilab

Массив - множественный тип данных, состоящий из фиксированного числа элементов.

Переменную, представляющую собой просто список данных, называют *одномерным массивом* или *вектором*. Для доступа к данным, хранящимся в определенном элементе массива, необходимо указать имя массива и порядковый номер этого элемента, называемый *индексом*.

Значение нижней границы индексации в Scilab равно единице. Индексы могут быть только целыми положительными числами.

Ввод и формирование одномерных массивов

Самый простой способ задать одномерный массив в Scilab имеет вид

$$[name] = X_n : dX : X_k,$$

Примечание: здесь и далее квадратные скобки (если не отмечено иначе) означают, что выражение, указанное в них может отсутствовать.

где *name* - имя переменной, в которую будет записан сформированный массив, X_n - значение первого элемента массива, X_k - значение последнего элемента массива, dX - шаг, с помощью которого формируется каждый следующий элемент массива, то есть значение второго элемента составит $X_n + dX$, третьего $X_n + dX + dX$ и т.д. до X_k .

Если параметр dX в конструкции отсутствует, это означает, что по умолчанию он принимает значение равное единице.

Переменную, заданную как массив, можно использовать в арифметических выражениях и в качестве аргумента математических функций.

Еще один способ задания векторов и матриц в Scilab - это их поэлементный ввод. Так, для *определения вектор-строки* следует ввести имя

массива, а затем после знака присваивания, в квадратных скобках через пробел или запятую перечислить элементы массива:

`[name]=x1 x2 ... xn` или `[name]=x1, x2, ..., xn`

Элементы *вектора-столбца* вводятся через точку с запятой:

`[name]=x1; x2; ...; xn`

Обратиться к элементу вектора можно, указав имя массива и порядковый номер элемента в круглых скобках:

`name (индекс)`

Функции в Scilab

Все *функции*, используемые в Scilab, можно разделить на два класса:

- *встроенные*;
- *определенные пользователем*.

В общем виде *обращение к функции* в Scilab имеет вид

`имя_переменной=имя_функции(переменная1, переменная2, ...)`

где `имя_переменной` - переменная, в которую будут записаны результаты работы функции; этот параметр может отсутствовать, тогда значение, вычисленное функцией будет присвоено системной переменной `ans`; `имя_функции` - имя встроенной функции или ранее созданной пользователем; `переменная1, переменная2, ...` - список аргументов функции.

Пакет Scilab снабжен достаточным количеством всевозможных встроенных функций. Например, элементарные математические функции, приведенные в указаниях к лабораторной работе 1.

Функции, определенные пользователем

Функция, как правило, предназначена для неоднократного использования, имеет входные параметры и не выполняется без их предварительного задания. Рассмотрим несколько способов *создания функций* в Scilab.

Первый способ это применение оператора `deff`, который в общем виде можно записать так:

`deff(' [имя1, ..., имяN]=имя_функции(переменная_1, ..., переменная_M) ', 'имя1=выражение1; ...; имяN=выражениеN')`

где `имя1, ..., имяN` - список выходных параметров, то есть переменных, которым будет присвоен конечный результат вычислений (парамет-

ров может быть от 1 до N); имя_функции – имя, с которым эта функция будет вызываться; переменная_1, ..., переменная_M - входные параметры (параметров может быть от 1 до M).

Второй способ создания функции – это применение конструкции вида:

```
function[имя1, ..., имяN]=имя_функции(переменная_1,  
..., переменная_M)  
    тело функции  
endfunction
```

где имя1, ..., имяN - список выходных параметров, то есть переменных, которым будет присвоен конечный результат вычислений (параметров может быть от 1 до N); имя_функции – имя, с которым эта функция будет вызываться, переменная_1, ..., переменная_M - входные параметры (параметров может быть от 1 до M).

Все имена переменных внутри функции, а также имена из списка входных и выходных параметров воспринимаются системой как *локальные*, то есть эти переменные считаются определенными только внутри функции.

Имя файла функции должно обязательно совпадать с именем функции. Расширение файлам-функциям обычно присваивают scf или sce.

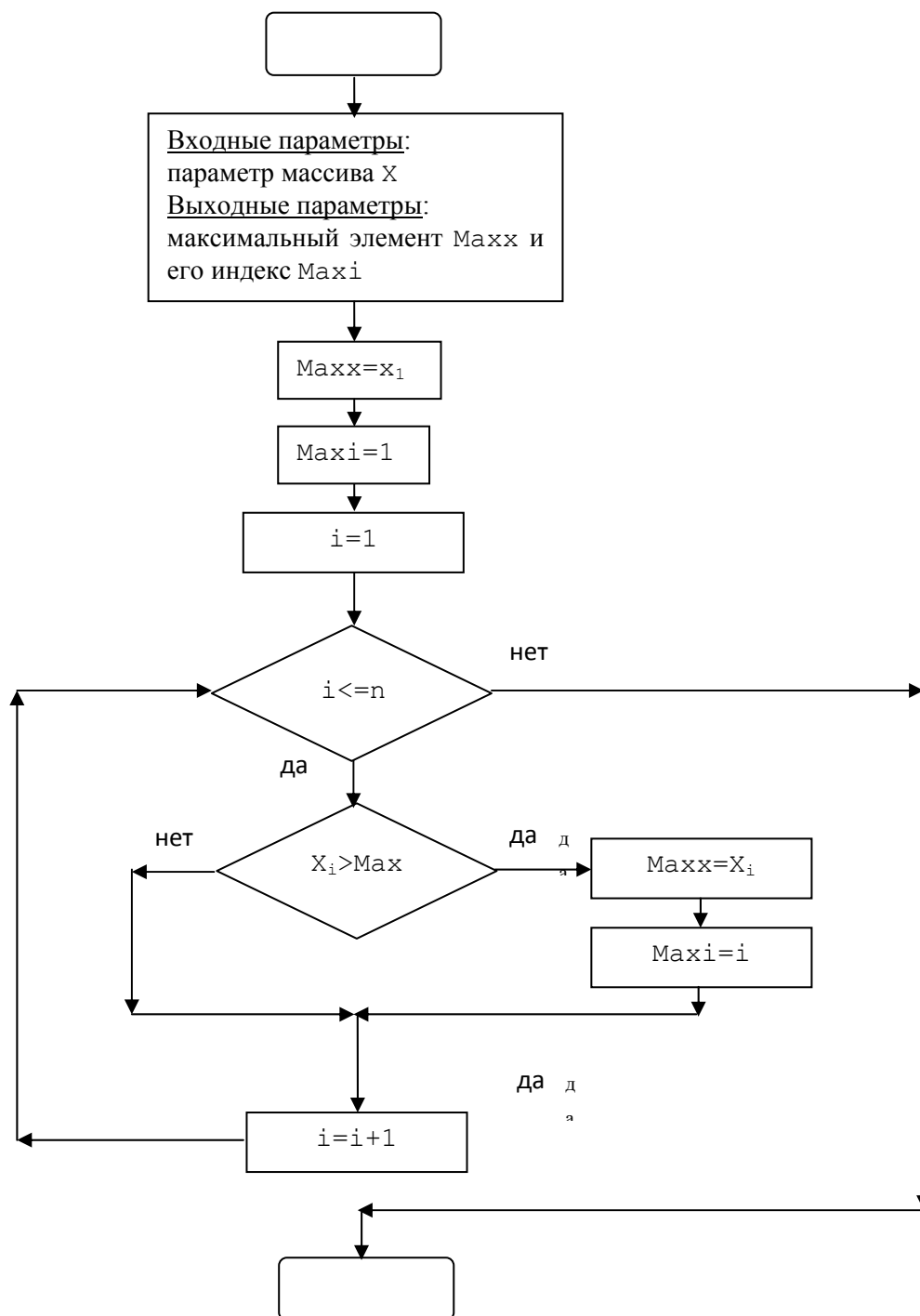
Обращение к функции осуществляется так же, как и к любой другой встроенной функции системы, то есть из командной строки. Однако функции, хранящиеся в отдельных файлах, должны быть предварительно загружены в систему, например при помощи оператора `ехес(имя_файла)` или командой главного меню «Файл → Выполнить», что, в общем, одно и то же.

Задачи

Задача 1. Написать функцию `maximal` для поиска максимального элемента и его индекса одномерного массива. Сравнить результат работы этой функции с результатом работы встроенной в Scilab функции `max`.

Блок-схема алгоритма

1. Блок-схема функции поиска максимума в массиве



Листинг программного кода

```
function [Max, Maxi]=maximal(X)
    Max=X(1);
    Maxi=1;
    n=length(X);
    for i=1:n
        if X(i)>Max then
            Max=X(i);
            Maxi=i;
        end
    end
endfunction
```

Результат работы программы

```
-->a=[3 2 5 4 7 5];
-->[el,num]=maximal(a)
num =
    5.
el =
    7.

-->[el,num]=max(a)
num =
    5.
el =
    7.
```

Результаты работы функции, определенной пользователем `maximal` и встроенной функции `max`, совпадают, что свидетельствует о правильности работы функции, определенной пользователем.

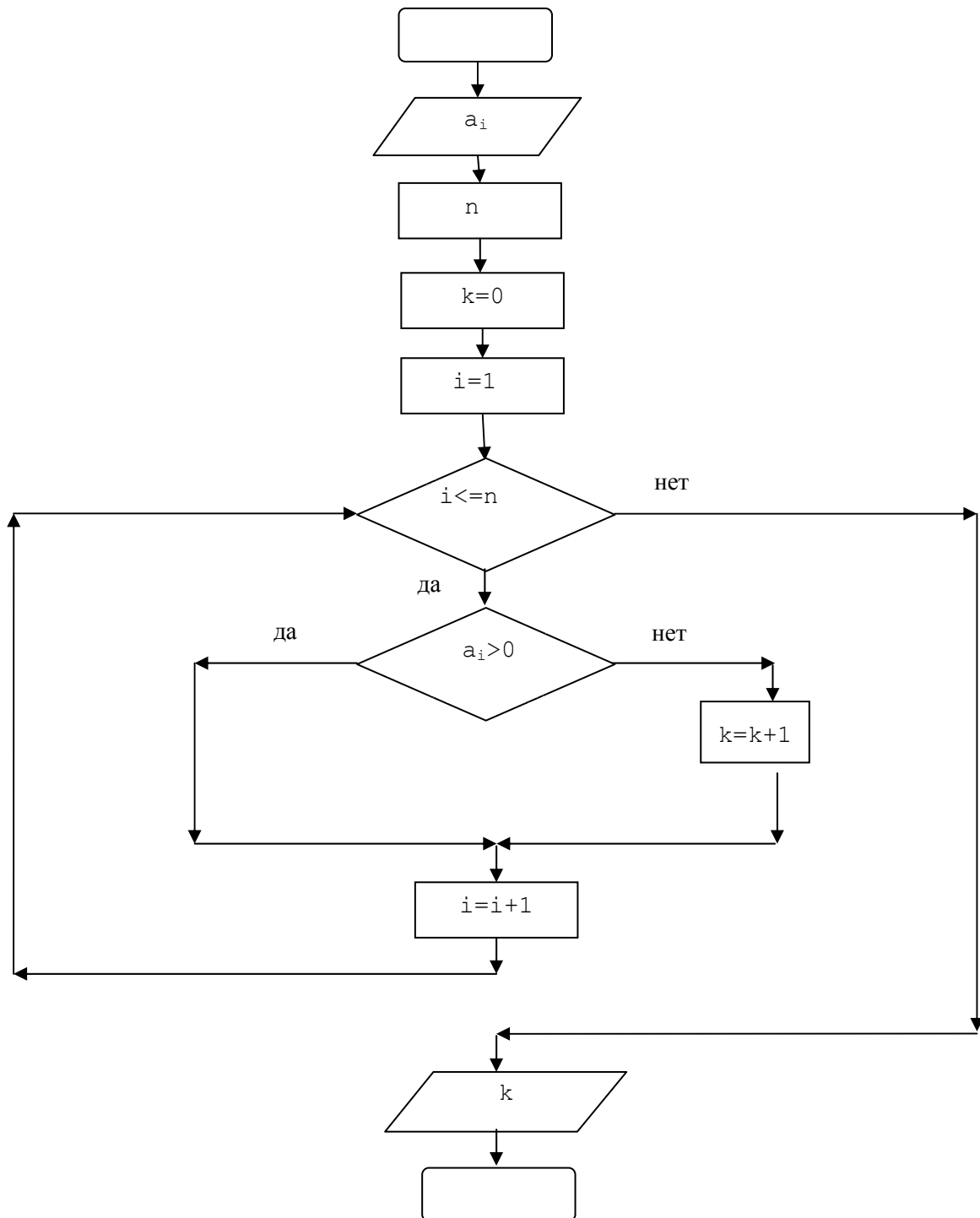
Задача 2. Написать функцию `minimal` для поиска минимального элемента и его индекса одномерного массива. Сравнить результат работы этой функции с результатом работы встроенной в Scilab функции `min`.

Задача 3. В одномерном массиве, состоящем из n целых элементов, вычислить

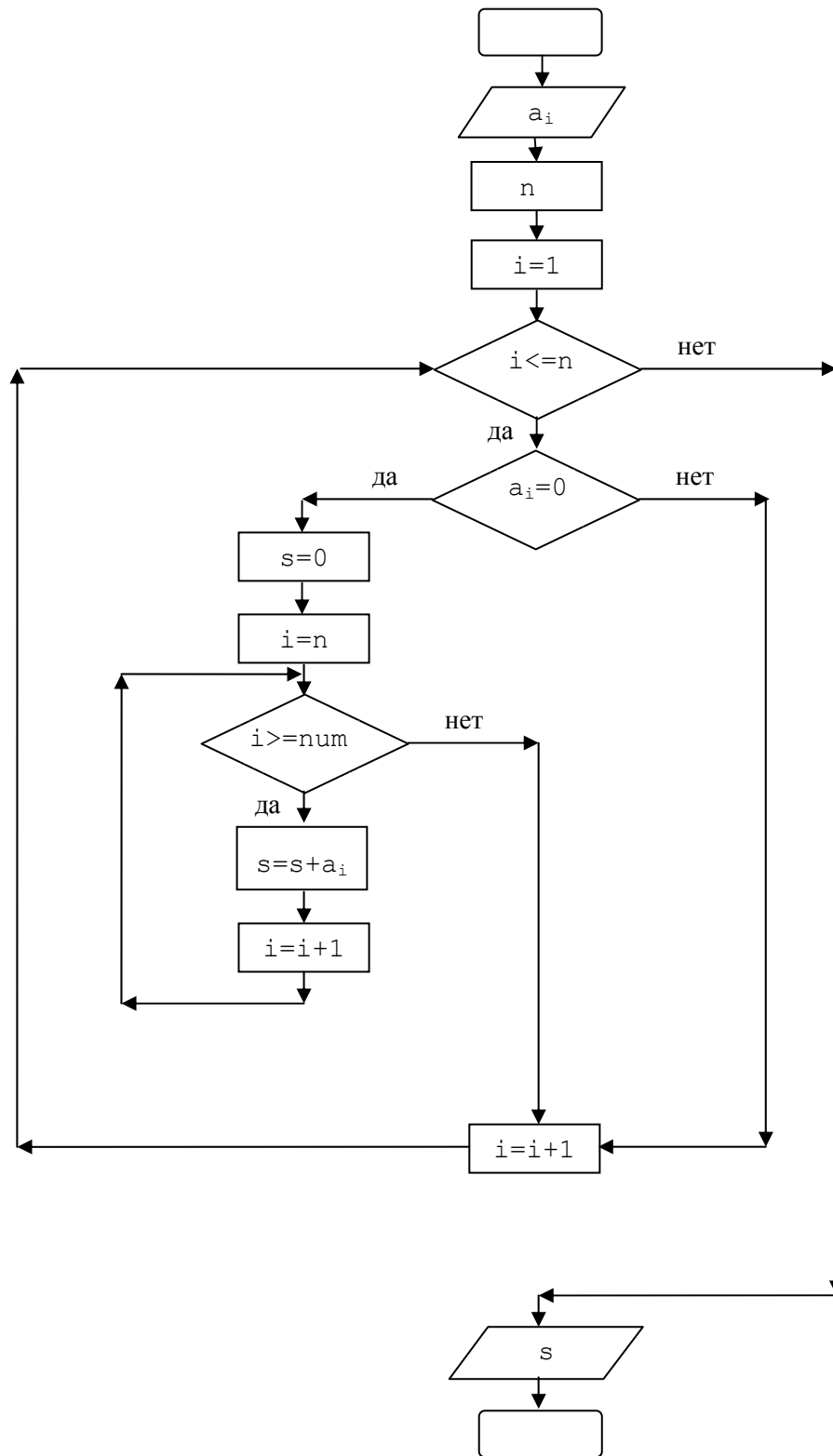
1. Количество положительных элементов массива.
2. Сумму элементов массива, расположенных после последнего элемента равного 0.
3. Преобразовать массив таким образом, чтобы сначала располагались все элементы, не превышающие единицу, а потом все остальные.

Блок-схема алгоритма

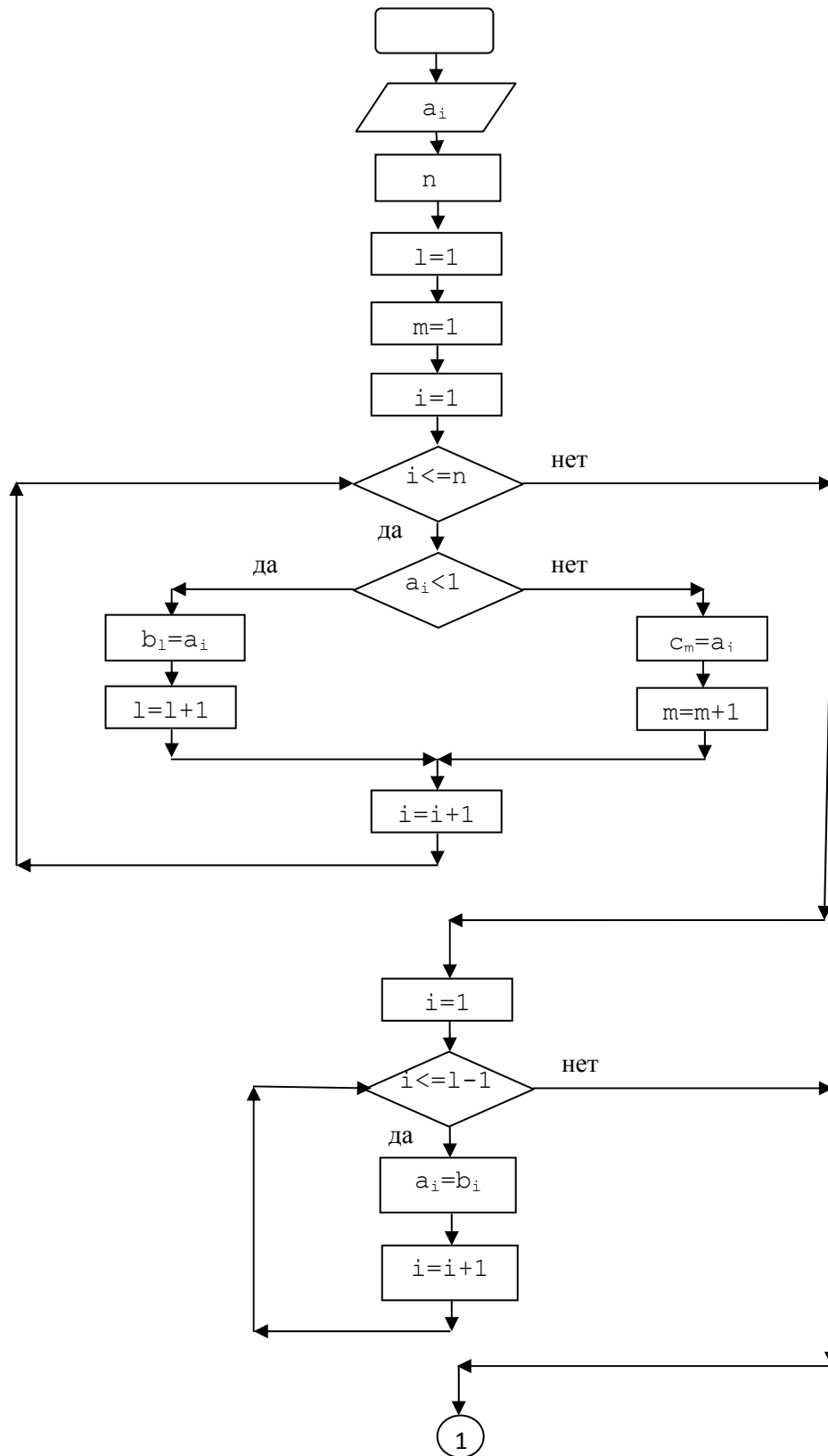
1. Блок-схема алгоритма поиска количества положительных элементов массива.

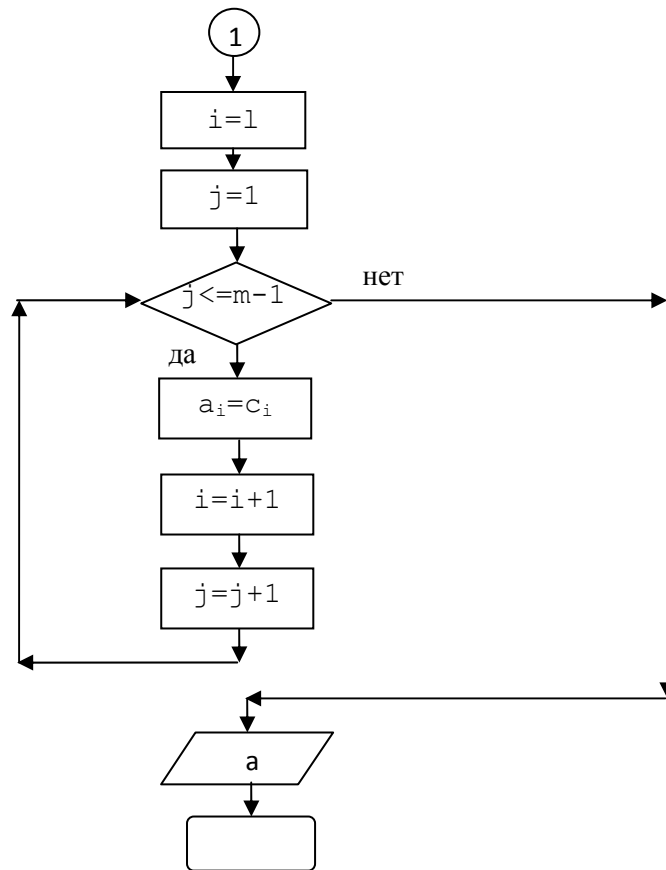


2. Блок-схема алгоритма поиска суммы элементов массива, расположенных после последнего элемента, равного 0.



3. Блок-схема алгоритма преобразования массива таким образом, чтобы сначала располагались все элементы, не превышающие единицу, а потом все остальные.





Листинг программного кода

```

n=10; //Задаем n
a=[-9 0 3 4 0 -5 1 -7 -5 10]; //Задаем массив из n
элементов
//Поиск кол-ва положительных элементов
k=0; //Задаем начальное значение кол-ва
for i=1:n //Перебираем все элементы массива
    if a(i)>0 then //Если элемент положительный
        k=k+1; //Увеличиваем кол-во на 1
    end
end
disp(a, 'a= '); //Выводим массив
disp(k, 'k= '); //Выводим кол-во
//Поиск суммы элементов массива после последнего 0
for i=1:n //Перебираем все элементы массива
    if a(i)==0 then //если элемент =0
        num=i; //то запоминаем его номер
        s=0; //задаем начальное значение суммы
        for i=n:-1:num //перебираем элементы с конца до
номера последнего эл-та=0
            s=s+a(i) //складываем эти элементы
        end
    end
end
disp(s, 's= ') //Выводим сумму
  
```

```

//Преобразование массива - сначала располагаются
//все элементы не превышающие 1, потом все остальные
l=1;//задаем начальное значение первого счетчика
m=1;//задаем начальное значение второго счетчика
for i=1:n do//перебираем все элементы массива a
    if a(i)<1 then//если элемент не превышает 1
        b(l)=a(i);//записываем его в новый массив b
        l=l+1//увеличиваем счетчик кол-ва элементов в
новом массиве b
    else//если элемент превышает 1
        c(m)=a(i);//записываем его в другой новый массив
с
        m=m+1//увеличиваем счетчик кол-ва эл-ов в другом
новом массиве с
    end
end
//Объединяем два новых массива,
//в b - элементы не превышающие 1
//в с - превышающие
for i=1:l-1//перебираем i от 1 до кол-ва элементов в
массиве b
    a(i)=b(i);//записываем элементы массива b в
преобразованный массив a
end
i=1;//задаем счетчик, который будет начинаться с 1 - кол-
ва эл-ов в массиве b
for j=1:m-1//перебираем j от 1 до кол-ва эл-ов в массиве
с
    a(i)=c(j);//записываем элементы массива с в
преобразованный массив a
    i=i+1//увеличиваем на 1 счетчик i
end
disp(a,'anew= ')//Вывод преобразованного массива a

```

Результат работы программы

```

-->exec('C:\Documents and Settings\Мои
документы\zad1_lab3.sce', -1)
a=
- 9.      0.      3.      4.      0. - 5.      1. - 7. - 5.
10.
k=
4.
s=
- 6.
anew=
- 9.      0.      0. - 5. - 7. - 5.      3.      4.      1.
10.

```

Задача 4. В одномерном массиве, состоящем из n целых элементов, вычислить

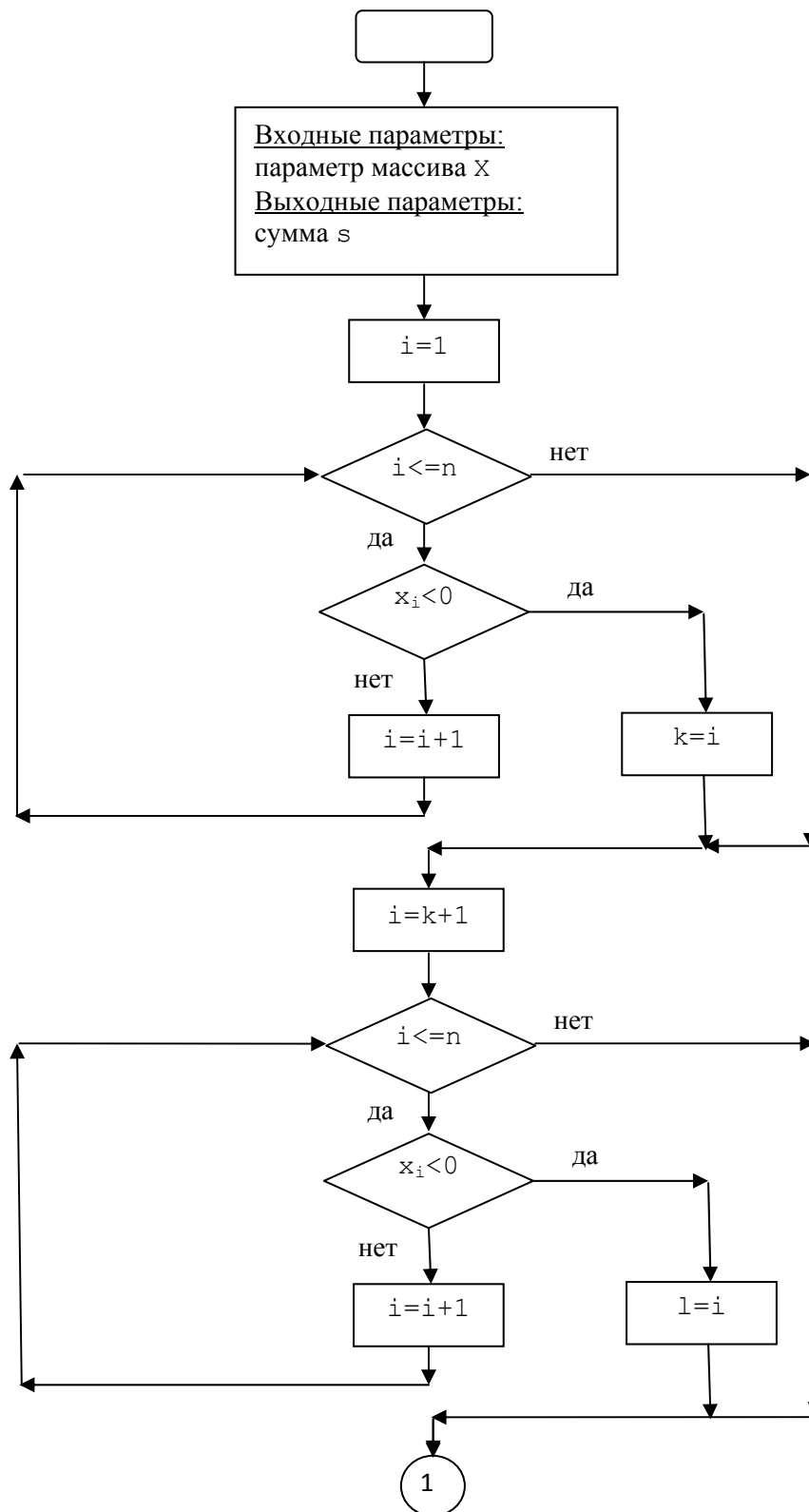
1. Сумму отрицательных элементов массива.
2. Произведение элементов массива, расположенных после последнего положительного элемента.
3. Преобразовать массив таким образом, чтобы сначала располагались все элементы, модуль которых не превышает 5, а потом все остальные.

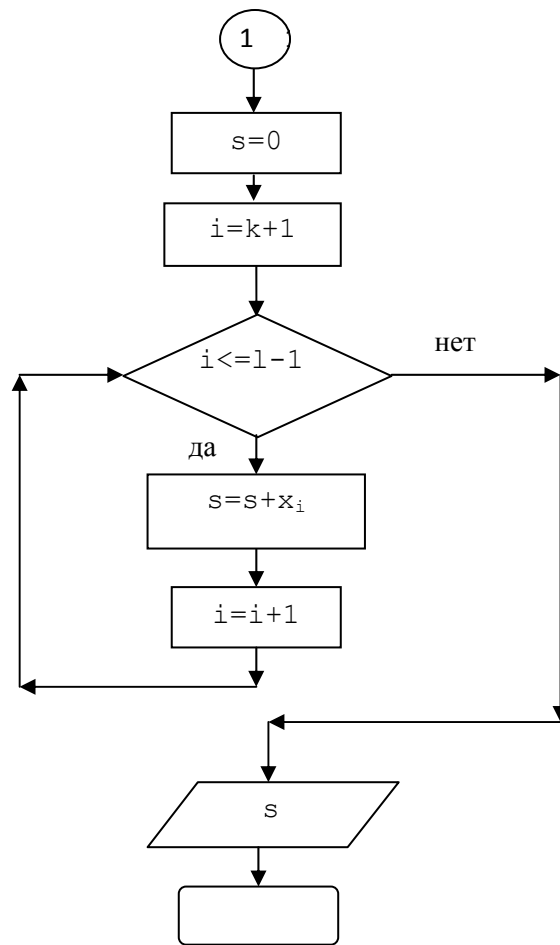
Задача 5. Дан одномерный массив, состоящий из n целых элементов.

1. Написать функцию поиска суммы элементов массива, расположенных между первым и вторым отрицательными элементами.
2. Написать функцию удаления числа с номером m из массива, где m вводится с клавиатуры.
3. Написать функцию для сортировки элементов массива по убыванию методом «пузырька». Сравнить результаты работы этой функции с результатом работы встроенной в Scilab функции `mtlb_sort`.

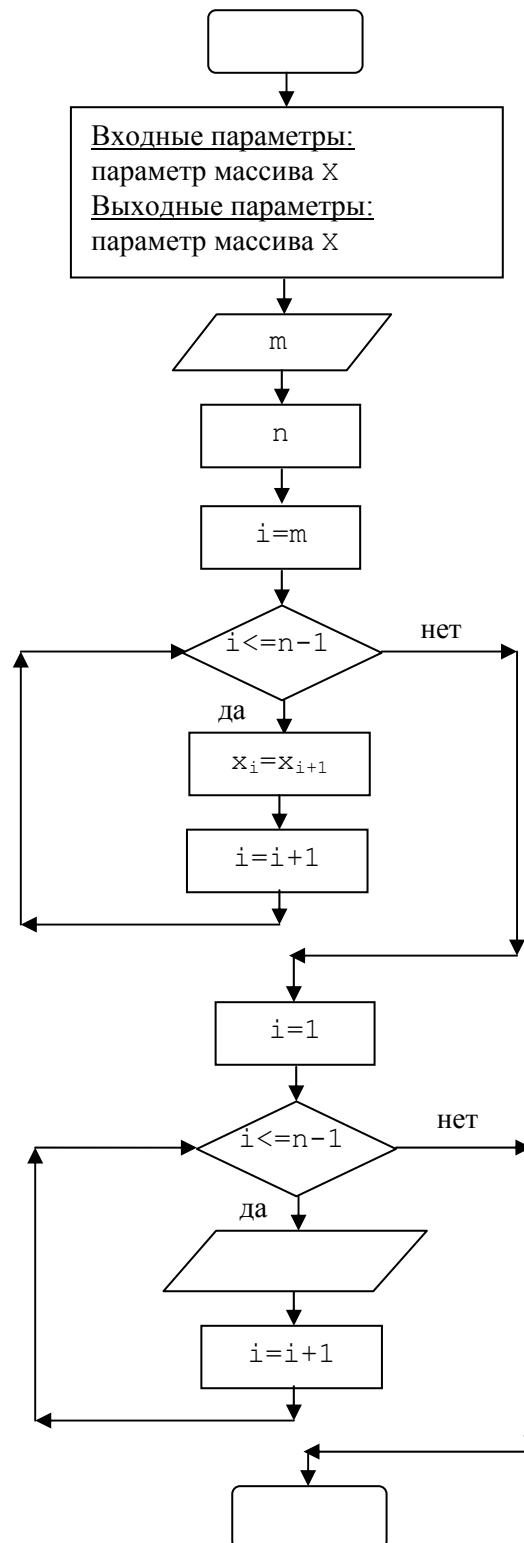
Блок-схема алгоритма

1. Блок-схема функции поиска суммы элементов массива, расположенных между первым и вторым отрицательными элементами

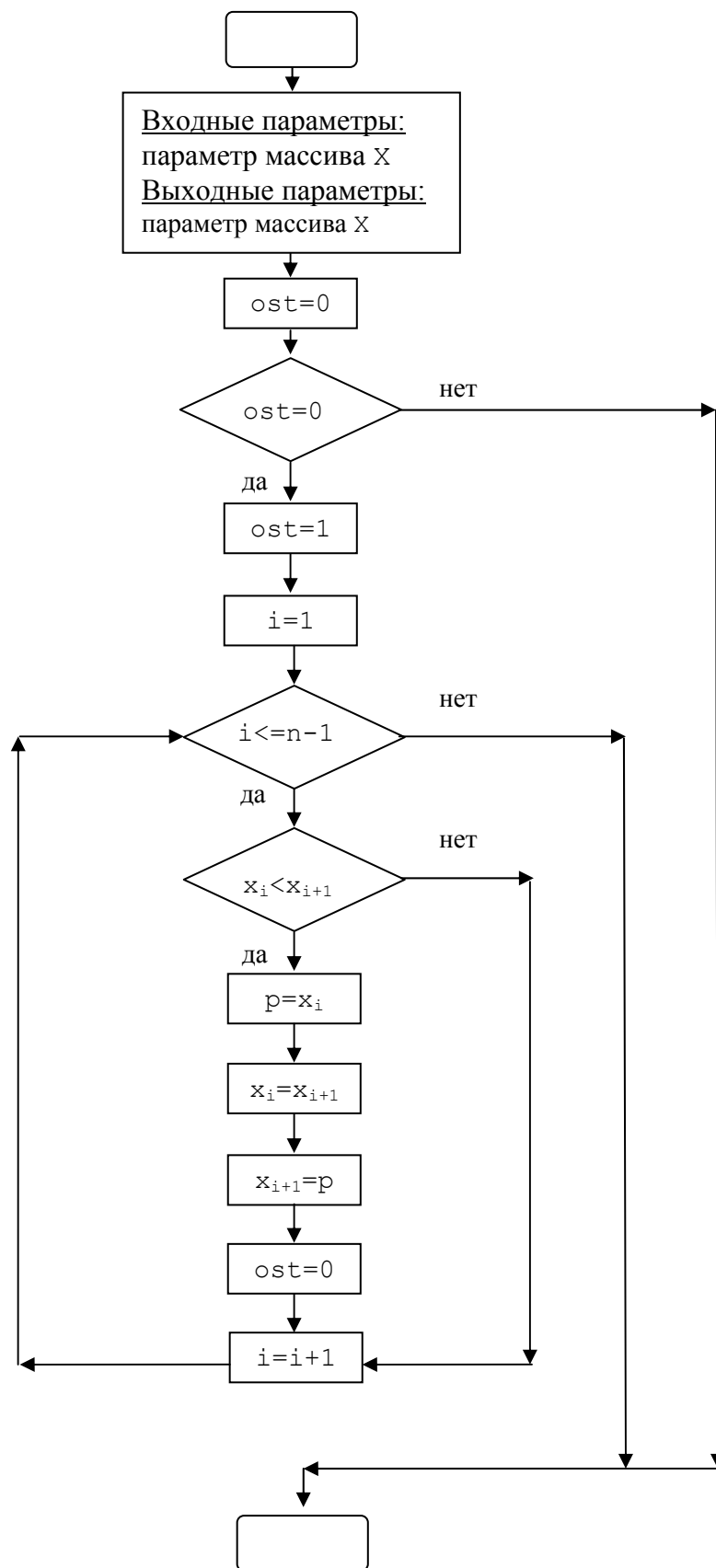




2. Блок-схема функции удаления числа с номером m из массива, где m вводится с клавиатуры.



3. Блок-схема функции для сортировки элементов массива по убыванию методом «пузырька».



Листинг программного кода

1.

```
function s=s_5_3(x)
    n=length(x);
    //Фиксируем номер первого отрицательного элемента
    for i=1:n
        if x(i)<0 then
            k=i;
            break
        end
    end
    //Фиксируем номер второго отрицательного элемента
    for i=k+1:n
        if x(i)<0 then
            l=i;
            break
        end
    end
    //Считаем сумму, перебирая элементы массива
    //от номера первого отрицательного элемента
    //до номера второго
    s=0;
    for i=k+1:l-1
        s=s+x(i)
    end
endfunction
```

2.

```
function x=udal_5_3(x)
    //Ввод номера удаляемого элемента.
    m=input('m=')
    n=length(x);
    //Сдвиг всех элементов, начиная с m-го на один влево.
    for i=m:n-1
        x(i)=x(i+1);
    end;
    //Удаление n-го элемента из массива.
    x(:,n)=[];
    //Уменьшение n на 1.
    n=n-1;
endfunction
```

3.

```
function x=sort_puz(x)
    n=length(x);
    ost=0;
```

```

while ost==0
    ost=1
    for i=1:n-1
        if x(i)<x(i+1) then
            p=x(i);
            x(i)=x(i+1);
            x(i+1)=p;
            ost=0
        end
    end
end
endfunction

```

Результат работы программы

```

-->exec('C:\Documents and Settings\Мои
документы\s_5_3.sci', -1)
-->a=[3 -1 2 7 0 -4 5 -4 1];summa=s_5_3(a)
summa =
    9.
-->exec('C:\Documents and Settings\Мои
документы\udal_5_3.sci', -1)
-->anew=udal_5_3(a)
m=3
anew =
    3. - 1.    7.    0. - 4.    5. - 4.    1.
-->exec('C:\Documents and Settings\Мои
документы\sort_puz.sci', -1)
-->anew1=sort_puz(a)
anew1 =
    7.    5.    3.    2.    1.    0. - 1. - 4. - 4.
-->-mtlb_sort(-a)
ans =
    7.    5.    3.    2.    1.    0. - 1. - 4. - 4.

```

Задача 6. Дан одномерный массив, состоящий из n целых элементов.

1. Написать функцию поиска произведения элементов массива, расположенных между первым и вторым положительными элементами.

2. Написать функцию удаления элементов, стоящих на нечетной позиции.

3. Написать функцию для сортировки элементов массива по возрастанию методом последовательных минимумов. Сравнить результаты работы этой функции с результатом работы встроенной в Scilab функции `mtlb_sort`.

Контрольные вопросы

1. Как в Scilab задать одномерный массив?

2. Нарисуйте блок-схему поиска максимального элемента в одномерном массиве. Какая существует встроенная в Scilab функция для поиска максимального элемента? Минимального?

3. Пусть в Scilab пользователем создана функция

```
function p=factorial(n)
    p=1
    for i=1:n
        p=p*i
    end
endfunction
```

Как называется функция? Где у функции входные аргументы и выходные значения? В какую переменную запишется результат?

4. Как в Scilab удалить элемент из одномерного массива?

5. Какие алгоритмы сортировки одномерных массивов вы знаете?

Какая существует встроенная в Scilab функция сортировки?

6. Пусть дан одномерный массив a из 10 элементов. Как обратиться к 5-му элементу этого массива?

7. Дан массив b

$b = [1 \ 4 \ 3 \ 2 \ 8 \ 7 \ 0]$

Что выведет Scilab при записи `disp(b(2))`?

8. Как работает функция `break`? Для чего она нужна?

9. Что произойдет в результате выполнения следующего кода программы в Scilab?

```
a=[1 2 3 4 5]
p=a(3)
a(3)=a(4)
a(4)=p
disp(a)
```

Зачем вводится переменная p ?

6. ПРОГРАММИРОВАНИЕ С ИСПОЛЬЗОВАНИЕМ ДВУМЕРНЫХ МАССИВОВ. ФУНКЦИИ

Цель работы: Приобретение практических навыков работы с двумерными массивами. Приобретение навыков работы в написании и отладке программ с использованием функций.

Постановка задачи: решить задачи с использованием двумерных массивов и функций.

Методические указания

Массивы и матрицы в Scilab. Решение задач линейной алгебры

Если возникает необходимость хранения данных в виде таблиц, в формате строк и столбцов, то необходимо использовать *двумерные массивы* (матрицы). Для доступа к данным, хранящимся в таком массиве, необходимо указать имя массива и два индекса, первый должен соответствовать номеру строки, а второй – номеру столбца, в которых хранится необходимый элемент.

Ввод и формирование матриц

Ввод элементов матрицы так же осуществляется в квадратных скобках, при этом элементы строки отделяются друг от друга пробелом или запятой, а строки разделяются между собой точкой с запятой:

```
[name]=[x11, x12, ..., x1n; x21, x22, ..., x2n;...; xm1, xm2, ..., xmn]
```

Обратиться к элементу матрицы можно, указав после имени матрицы, в круглых скобках, через запятую, номер строки и номер столбца на пересечении которых элемент расположен:

```
name(индекс1, индекс2)
```

Кроме того, матрицы и векторы можно формировать, составляя их из ранее заданных матриц и векторов.

Важную роль при работе с матрицами играет знак двоеточия «:». Указывая его вместо индекса при обращении к массиву, можно иметь доступ к группам его элементов.

Действия над матрицами

Для работы с матрицами и векторами в Scilab предусмотрены следующие операции:

- + сложение;
- вычитание;
- ’ транспонирование;
- * матричное умножение;
- * умножение на число;

^	возведение в степень;
\	левое деление;
/	правое деление;
*	поэлементное умножение матриц;
.^	поэлементное возведение в степень;
.\	поэлементное левое деление;
./	поэлементное правое деление.

Также для работы с матрицами и векторами в Scilab существуют специальные матричные функции, функции вычисления специальных числовых характеристик матриц и векторов, функции, реализующие численные алгоритмы решения задач линейной алгебры. С этими функциями можно подробно познакомиться, например в [3].

Задачи

Задача 1. Решить систему линейных алгебраических уравнений

$$\begin{cases} 2x_1 + x_2 - 5x_3 + x_4 = 8 \\ x_1 - 3x_2 - 6x_4 = 9 \\ 2x_2 - x_3 + 2x_4 = -5 \\ x_1 + 4x_2 - 7x_3 + 6x_4 = 0 \end{cases}$$

1. Методом Крамера, сделать проверку;
2. Методом обратной матрицы;
3. Методом Гаусса.

Решение

1. *Правило Крамера.* Если определитель $\Delta = \det A$ матрицы системы из n уравнений с n неизвестными $Ax=b$ отличен от нуля, то система имеет единственное решение $x_1, x_2, \dots, x_n$, определяемое по формулам Крамера: $x_i = \Delta_i / \Delta$, где Δ_i - определитель матрицы, полученной из матрицы системы A заменой i -го столбца столбцом свободных членов b .

2. *Метод обратной матрицы:* для системы из n линейных уравнений с n неизвестными $Ax=b$, при условии, что определитель матрицы A не равен нулю, единственное решение можно представить в виде $x = A^{-1}b$;

3. Решение системы линейных уравнений при помощи *метода Гаусса* основывается на том, что от заданной системы переходят к эквивалентной системе, которая решается проще, чем исходная система. Метод Гаусса состоит из двух этапов. Первый этап - это прямой ход, в результате которого расширенная матрица системы путем элементарных преобразований (перестановка уравнений системы, умножение уравнений на число, отличное от нуля, и сложение уравнений) приводится к

ступенчатому виду. На втором этапе (обратный ход) ступенчатую матрицу преобразовывают так, чтобы в первых n столбцах получилась единичная матрица. Последний, $n+1$, столбец этой матрицы содержит решение системы линейных уравнений.

Листинг программного кода

```
1.
//Матрица коэффициентов:
A=[2 1 -5 1;1 -3 0 -6;0 2 -1 2;1 4 -7 6];
b=[8;9;-5;0]; //Вектор свободных коэффициентов
A1=A;A1(:,1)=b; //Первая вспомогательная матрица
A2=A;A2(:,2)=b; //Вторая вспомогательная матрица
A3=A;A3(:,3)=b; //Третья вспомогательная матрица
A4=A;A4(:,4)=b; //Четвертая вспомогательная матрица
D=det(A); //Главный определитель
//Определители вспомогательных матриц:
d(1)=det(A1); d(2)=det(A2); d(3)=det(A3); d(4)=det(A4);
x=d/D; //Вектор неизвестных
P=A*x-b; //Проверка
disp(x, 'x='); //Вывод x
disp(P, 'P='); //Вывод P
```

```
2.
//Матрица и вектор свободных коэффициентов системы:
A=[2 1 -5 1;1 -3 0 -6;0 2 -1 2;1 4 -7 6];
b=[8;9;-5;0];
x=inv(A)*b //Решение системы
disp(x, 'x=') //Вывод x
```

```
3.
//Матрица и вектор свободных коэффициентов системы:
//A=[2 -1 1;3 2 -5;1 3 -2]; b=[0;1;4];
A=[2 1 -5 1;1 -3 0 -6;0 2 -1 2;1 4 -7 6];b=[8;9;-5;0];
//Приведение расширенной матрицы к треугольному виду:
C=rref([A b]);
//Определение размерности расширенной матрицы:
[n,m]=size(C); //m- номер последнего столбца матрицы C
//Выделение последнего столбца из матрицы C:
x=C(:,m) //x - решение системы
disp(x, 'x=') //Вывод x
```

Результат работы программы

```
1.
-->exec('C:\Documents and Settings\Мои
документы\work_Scilab\lab4_1_kramer.sce', -1)
x=
3.
```

```

- 4.
- 1.
  1.
P=
  0.
  0.
- 8.882D-16
  2.665D-15

```

2.

```

-->exec('C:\Documents          and          Settings\Мои
документы\work_Scilab\lab4_1_obrmatr.sce', -1)

```

x=

```

  3.
- 4.
- 1.
  1.

```

3.

```

-->exec('C:\Documents          and          Settings\Мои
документы\work_Scilab\lab4_1_gauss.sce', -1)

```

x=

```

  3.
- 4.
- 1.
  1.

```

Задача 2. Решить систему линейных алгебраических уравнений, сделать проверку (по вариантам).

1. $\begin{cases} -x_1 - x_2 - 2x_3 - 3x_4 = 2 \\ 3x_1 - x_2 - x_3 - 2x_4 = -8 \\ 2x_1 + 3x_2 - x_3 - x_4 = -12 \\ x_1 + 2x_2 + 3x_3 - x_4 = 8 \end{cases}$	2. $\begin{cases} x_1 - 2x_2 + 3x_3 - 2x_4 = -6 \\ x_1 + x_2 - 2x_3 - 3x_4 = -8 \\ 3x_1 - 2x_2 - x_3 + 2x_4 = 4 \\ 2x_1 + 3x_2 + 2x_3 + x_4 = 8 \end{cases}$
3. $\begin{cases} x_1 + 2x_2 + 3x_3 + 3x_4 = 5 \\ 2x_1 + x_2 + 2x_3 + 3x_4 = 1 \\ 3x_1 + 2x_2 + x_3 + 2x_4 = 1 \\ 4x_1 + 3x_2 + 2x_3 + x_4 = -5 \end{cases}$	4. $\begin{cases} 0.1x_1 + 0.5x_2 + 0.3x_3 - 0.4x_4 = 2 \\ 0.3x_1 + 0.1x_2 - 0.2x_3 = 0.9 \\ 0.5x_1 - 0.7x_2 + 1x_4 = -0.9 \\ 0.3x_2 - 0.5x_3 = 0.1 \end{cases}$

5. $\begin{cases} 10x_2 + 30x_3 + 40x_4 = -50 \\ 10x_1 + 20x_3 + 30x_4 = -40 \\ 30x_1 + 20x_2 - 50x_4 = 120 \\ 40x_1 + 30x_2 + 50x_3 = 50 \end{cases}$	6. $\begin{cases} 0.3x_1 + x_2 + 1.67x_3 - 2.3x_4 = 4 \\ 3x_1 + 5x_2 + 7x_3 - x_4 = 0 \\ 5x_1 + 7x_2 + x_3 - 3x_4 = 4 \\ 7x_1 + x_2 + 3x_3 - 5x_4 = 16 \end{cases}$
7. $\begin{cases} 2x_1 + x_2 + 5x_3 + x_4 = 8 \\ 0.333x_1 - x_2 - 2x_4 = 3 \\ 2x_2 + x_3 + 2x_4 = -5 \\ x_1 + 4x_2 + 7x_3 + 6x_4 = 0 \end{cases}$	8. $\begin{cases} -x_1 + x_2 + x_3 + x_4 = 12 \\ 2x_1 + x_2 + 2x_3 + 3x_4 = 13 \\ 1.5x_1 + x_2 + 0.5x_3 + x_4 = 7 \\ 4x_1 + 3x_2 + 2x_3 + x_4 = -15 \end{cases}$
9. $\begin{cases} -2x_1 - x_2 + 3x_3 + 2x_4 = 40 \\ -x_1 + x_2 + x_3 + 0.6667x_4 = 20 \\ -3x_1 - x_2 - x_3 + 2x_4 = 60 \\ -3x_1 - x_2 + 3x_3 - x_4 = 60 \end{cases}$	10. $\begin{cases} 3x_1 - 6x_2 - 3x_3 + 3x_4 = 8 \\ 2x_1 - x_2 + x_3 + x_4 = 5 \\ x_1 + x_2 + 2x_3 + x_4 = -1 \\ x_1 - x_2 - x_3 + 3x_4 = 10 \end{cases}$
11. $\begin{cases} 20x_1 + 5x_2 + 5x_4 = -9 \\ x_1 - 3x_2 + 4x_3 = -7 \\ 3x_2 - 2x_3 - 4x_4 = 12 \\ x_1 + 2x_2 - x_3 + 3x_4 = 10 \end{cases}$	12. $\begin{cases} x_1 - 3x_2 + x_3 + x_4 = 11 \\ x_1 + 3x_2 + 5x_3 + 7x_4 = 12 \\ 3x_1 + 5x_2 + 7x_3 + x_4 = 0 \\ -5x_1 - 7x_2 - x_3 - 3x_4 = -4 \end{cases}$
13. $\begin{cases} 2x_1 + x_2 + x_3 - x_4 = 11 \\ 2x_1 + x_2 - 3x_4 = 2 \\ 3x_1 + x_3 + x_4 = -3 \\ 4x_1 - 4x_2 - 4x_3 + 10x_4 = 7 \end{cases}$	14. $\begin{cases} 2x_1 + x_3 + 4x_4 = 19 \\ x_1 + 2x_2 - x_3 + x_4 = 18 \\ 2x_1 + x_2 + x_3 + x_4 = 15 \\ 2x_1 - 2x_2 + 4x_3 + 2x_4 = -11 \end{cases}$
15. $\begin{cases} 5x_1 - 3x_2 + x_3 + x_4 = 11 \\ x_1 + 3x_2 + 5x_3 + 7x_4 = 12 \\ 3x_1 + 5x_2 + 7x_3 + x_4 = 0 \\ -5x_1 - 7x_2 - x_3 - 3x_4 = -4 \end{cases}$	

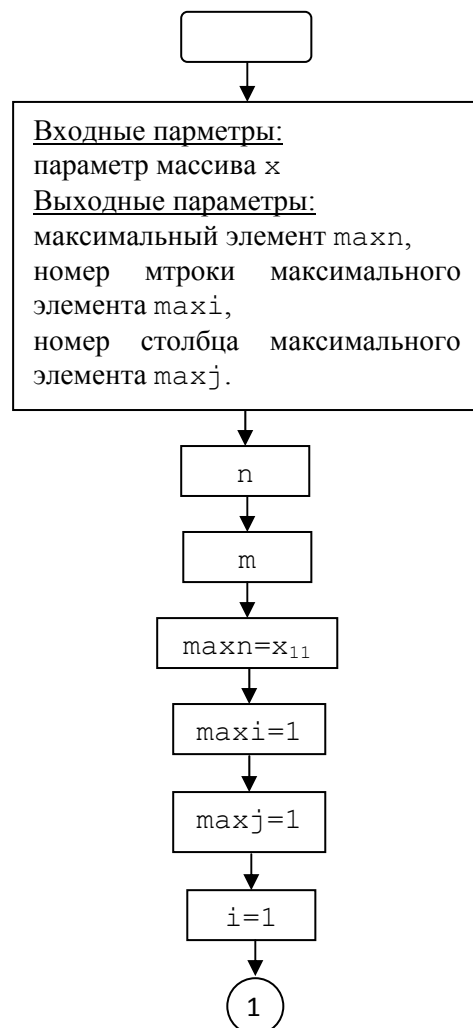
Задача 3. Если возможно, вычислить матрицу, обратную к матрице D (по вариантам).

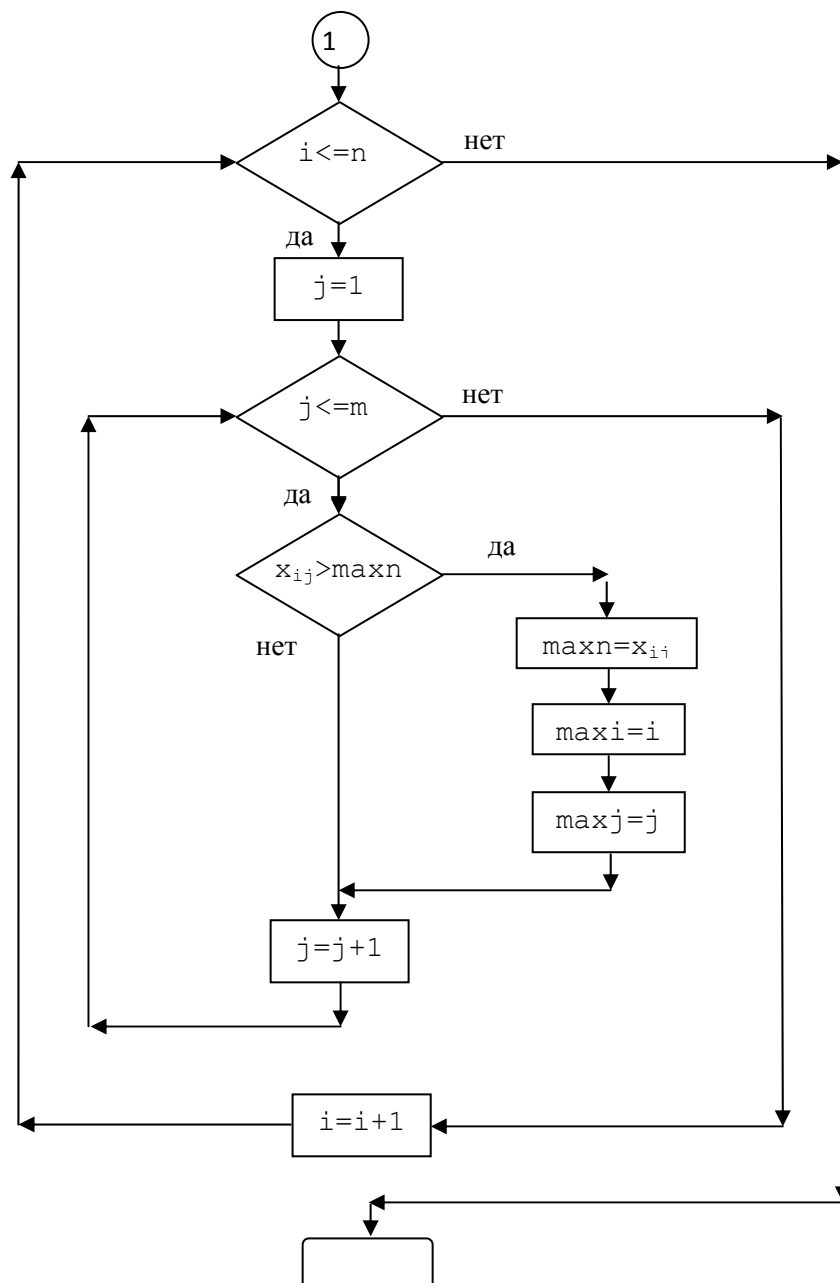
1. $D = 2(A^2 + B)(2B - A)$, где $A = \begin{pmatrix} 2 & 3 & -1 \\ 4 & 5 & 2 \\ -1 & 0 & 7 \end{pmatrix}$, $B = \begin{pmatrix} -1 & 0 & 5 \\ 0 & 1 & 3 \\ 2 & -2 & 4 \end{pmatrix}$	2. $D = 3A - (A + 2B)B^2$, где $A = \begin{pmatrix} 4 & 5 & -2 \\ 3 & -1 & 0 \\ 4 & 2 & 7 \end{pmatrix}$, $B = \begin{pmatrix} 2 & 1 & -1 \\ 0 & 1 & 3 \\ 5 & 7 & 3 \end{pmatrix}$
3. $D = 3A^2 - (A + 2B)B$, где $A = \begin{pmatrix} 4 & 5 & -2 \\ 3 & -1 & 0 \\ 4 & 2 & 7 \end{pmatrix}$, $B = \begin{pmatrix} 2 & 1 & -1 \\ 0 & 1 & 3 \\ 5 & 7 & 3 \end{pmatrix}$	4. $D = (A - B^2)2(2A + B^3)$, где $A = \begin{pmatrix} 5 & 2 & 0 \\ 10 & 4 & 1 \\ 7 & 3 & 2 \end{pmatrix}$, $B = \begin{pmatrix} 3 & 6 & -1 \\ -1 & -2 & 0 \\ 2 & 1 & 3 \end{pmatrix}$
5. $D = 2(A - B)(A^2 + B)$, где $A = \begin{pmatrix} 5 & 1 & 7 \\ -10 & -2 & 1 \\ 0 & 1 & 2 \end{pmatrix}$, $B = \begin{pmatrix} 2 & 4 & 1 \\ 3 & 1 & 0 \\ 7 & 2 & 1 \end{pmatrix}$	6. $D = (A - B)^2 A + 2B$, где $A = \begin{pmatrix} 5 & -1 & 3 \\ 0 & 2 & -1 \\ -2 & -1 & 0 \end{pmatrix}$, $B = \begin{pmatrix} 3 & 7 & -2 \\ 1 & 1 & -2 \\ 0 & 1 & 3 \end{pmatrix}$
7. $D = (A^2 - B^2)(A - B^3)$, где $A = \begin{pmatrix} 7 & 2 & 0 \\ -7 & -2 & 1 \\ 1 & 1 & 1 \end{pmatrix}$, $B = \begin{pmatrix} 0 & 2 & 3 \\ 1 & 0 & -2 \\ 3 & 1 & 1 \end{pmatrix}$	8. $D = 2(A - B)(A^2 + B)$, где $A = \begin{pmatrix} 5 & 1 & 7 \\ -10 & -2 & 1 \\ 0 & 1 & 2 \end{pmatrix}$, $B = \begin{pmatrix} 2 & 4 & 1 \\ 3 & 1 & 0 \\ 7 & 2 & 1 \end{pmatrix}$
9. $D = 2A - (A^2 + B)B$, где $A = \begin{pmatrix} 1 & 4 & 2 \\ 2 & 1 & -2 \\ 0 & 1 & -1 \end{pmatrix}$, $B = \begin{pmatrix} 4 & 6 & -2 \\ 4 & 10 & 1 \\ 2 & 4 & -5 \end{pmatrix}$	10. $D = 2(A - 0.5B) + A^3 B$, где $A = \begin{pmatrix} 5 & 3 & -1 \\ 2 & 0 & 4 \\ 3 & 5 & -1 \end{pmatrix}$, $B = \begin{pmatrix} 1 & 4 & 16 \\ -3 & -2 & 0 \\ 5 & 7 & 2 \end{pmatrix}$
11. $D = (A - B)A^2 + 3B$, где $A = \begin{pmatrix} 3 & 2 & -5 \\ 4 & 2 & 0 \\ 1 & 1 & 2 \end{pmatrix}$, $B = \begin{pmatrix} -1 & 2 & 4 \\ 0 & 3 & 2 \\ -1 & -3 & 4 \end{pmatrix}$	12. $D = 3(A^2 + B^2) - 2AB$, где $A = \begin{pmatrix} 4 & 2 & 1 \\ 3 & -2 & 0 \\ 0 & -1 & 2 \end{pmatrix}$, $B = \begin{pmatrix} 2 & 0 & 2 \\ 5 & -7 & -2 \\ 1 & 0 & -1 \end{pmatrix}$

13. $D = 2A^3 + 3B(AB - 2A)$, где $A = \begin{pmatrix} 1 & -1 & 0 \\ 2 & 0 & -1 \\ 1 & 1 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 5 & 3 & 1 \\ -1 & 2 & 0 \\ -3 & 0 & 0 \end{pmatrix}$	14. $D = A(A^2 - B) - 2(B + A)B$, где $A = \begin{pmatrix} 2 & 3 & 1 \\ -1 & 2 & 4 \\ 5 & 3 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 7 & 13 \\ -1 & 0 & 5 \\ 5 & 13 & 21 \end{pmatrix}$
15. $D = (2A - B)(3A + B) - 2A^2B$, где $A = \begin{pmatrix} 1 & 0 & 3 \\ -2 & 0 & 1 \\ -1 & 3 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 7 & 5 & 2 \\ 0 & 1 & 2 \\ -3 & -1 & -1 \end{pmatrix}$	

Задача 4. Написать функцию `maximaln` для поиска максимального элемента и его индексов двумерного массива. Сравнить результат работы этой функции с результатом работы встроенной в Scilab функции `max`.

Блок-схема алгоритма





Листинг программного кода

```
function [Maxn, Maxi, Maxj]=maximaln(x)
    n=size(x,1);
    m=size(x,2);
    Maxn=x(1,1);
    Maxi=1;
    Maxj=1;
    for i=1:n
        for j=1:m
            if x(i,j)>Maxn then
                Maxn=x(i,j);
                Maxi=i;
                Maxj=j;
            end
        end
    end
endfunction
```

Результат работы программы

```
-->x=[1 4 3;9 4 5;-8 7 0];[el,numi,numj]=maximaln(x)
numj =
    1.
numi =
    2.
el =
    9.
-->x=[1 4 3;9 4 5;-8 7 0];[el,num]=max(x)
num =

    2.    1.
el =

    9.
```

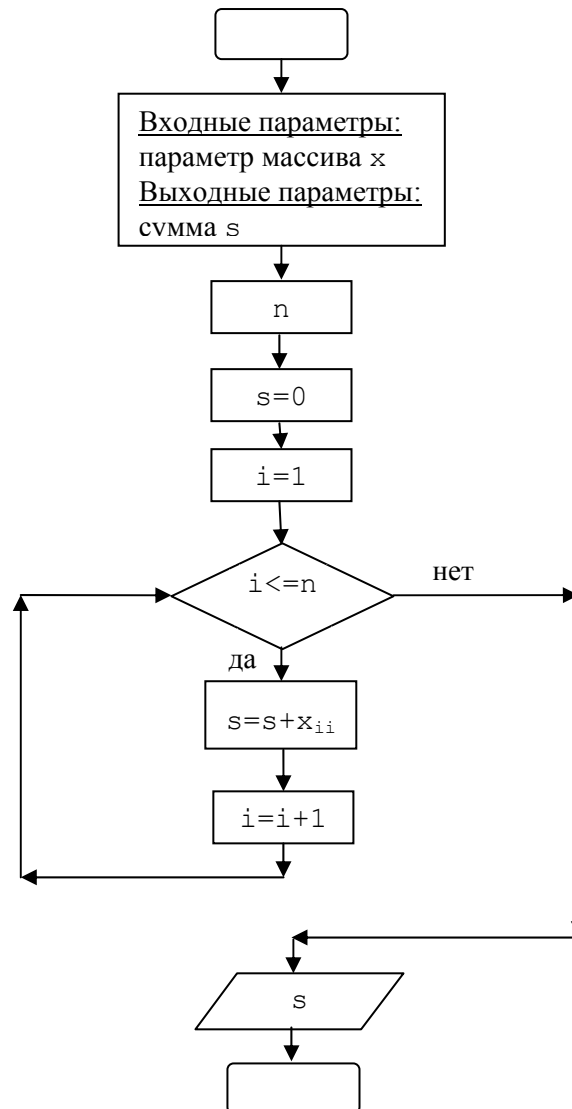
Задача 5. Написать функцию `minimaln` для поиска минимального элемента и его индексов двумерного массива. Сравнить результат работы этой функции с результатом работы встроенной в Scilab функции `min`.

Задача 6. Дана квадратная матрица A .

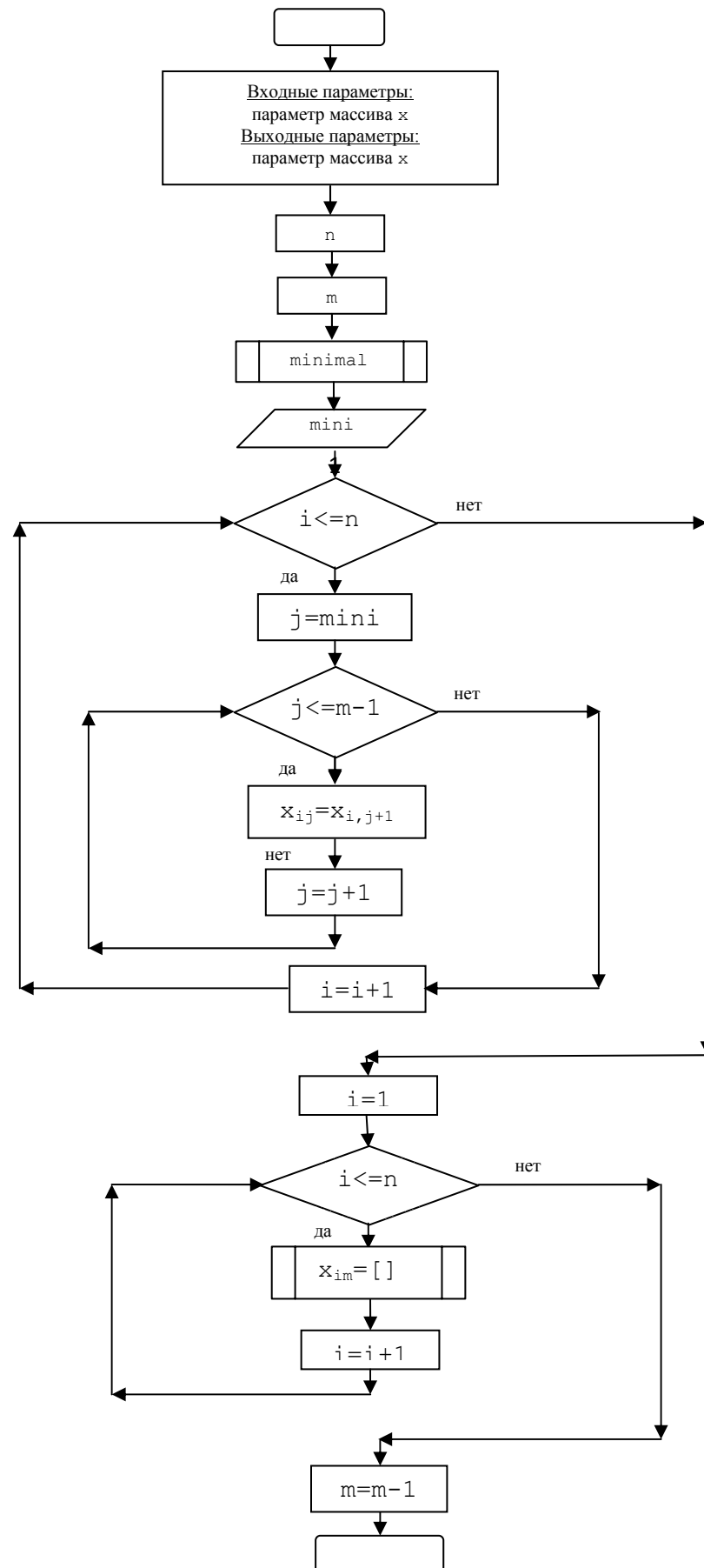
1. Написать функцию поиска суммы элементов в главной диагонали;
2. Написать функцию удаления столбца с минимальным элементом (для поиска минимального элемента и его номера рекомендуется обращение к функции `minimaln`);
3. Написать функцию поиска произведения всех элементов первого столбца.

Блок-схема алгоритма

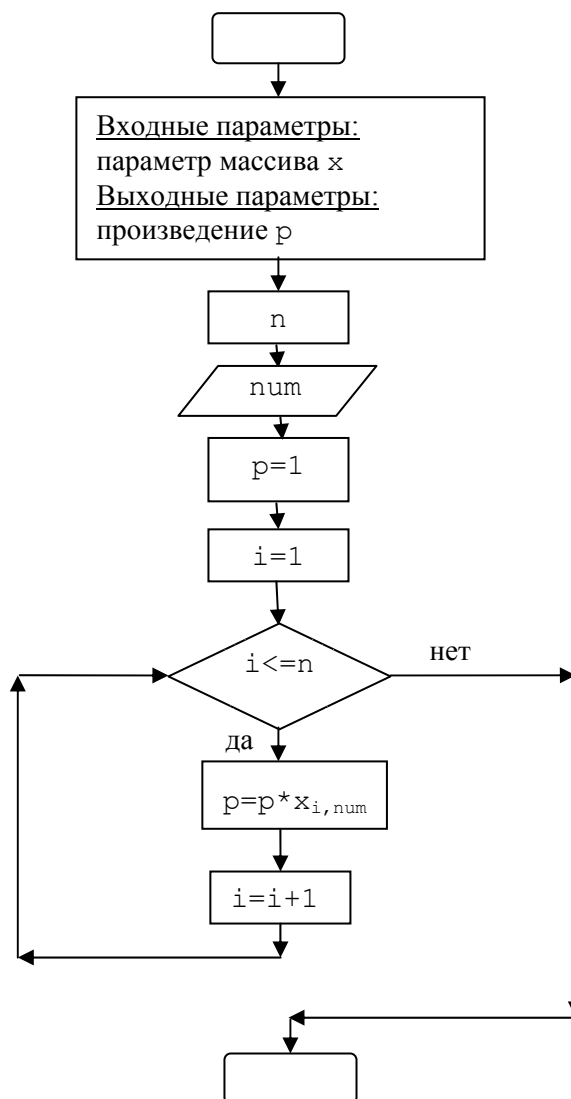
1. Блок-схема функции поиска суммы элементов на главной диагонали матрицы



2. Блок-схема функции удаления столбца с минимальным элементом в матрице.



3. Блок-схема поиска произведения всех элементов первого столбца.



Листинг программного кода

1.

```

function s=diagonal(x)
    n=size(x,1); //Определяем кол-во строк и столбцов в массиве x
    s=0; //Задаем начальное значение суммы
    for i=1:n
        s=s+x(i,i); //Складываем элементы на глав
диагонали
    end
endfunction
  
```

2.

```

function x=udal_stolbca(x)
    n=size(x,1); //Определяем кол-во строк в массиве x
  
```

```

m=size(x,2); //Определяем кол-во столбцов в массиве x
//Обращение к функции поиска минимального
[minn,mini]=minimaln(x);
disp(mini,'mini=');
//Сдвиг всех элементов, начиная с m-го на один влево.
for i=1:n
    for j=mini:m-1
        x(i,j)=x(i,j+1);
    end
end;
// Удаление m-го столбца из массива.
x(:,m)=[];
//Уменьшение m на 1.
m=m-1;
endfunction

```

3.

```

function p=proizv_stolbca(x)
    n=size(x,1); //Определяем кол-во строк в массиве x
    num=input('num= ') //Вводим номер перемножаемой строки
    p=1; //Задаем начальное значение произведения
    for i=1:n
        p=p*x(i,num); //Перемножаем элементы в столбце
    end
endfunction

```

с номером num

Результат работы программы

```

-->exec('C:\Documents and Settings\Мои
документы\work_Scilab\diagonal.sci', -1)

```

```

-->exec('C:\Documents and Settings\Мои
документы\work_Scilab\minimal.sci', -1)

```

```

-->exec('C:\Documents and Settings\Мои
документы\work_Scilab\udal_stolbca.sci', -1)

```

```

-->exec('C:\Documents and Settings\Мои
документы\work_Scilab\proizv_stolbca.sci', -1)

```

```

-->x=[6 -3 4 5;-4 5 3 6;9 -4 5 6;3 4 1 2]
x =
    6.   -3.    4.    5.
   -4.    5.    3.    6.
    9.   -4.    5.    6.
    3.    4.    1.    2.

```

```
-->s=diagonal(x)
s =
    18.

-->xnew=udal_stolbca(x)
mini=
    2.
xnew =
    6.    4.    5.
- 4.    3.    6.
    9.    5.    6.
    3.    1.    2.

-->p=proizv_stolbca(x)
num= 1
p =
- 648.
```

Задача 7. Дана квадратная матрица A .

1. Написать функцию поиска суммы элементов на диагонали ниже/выше главной;
4. Написать функцию удаления столбца с максимальным элементом (для поиска максимального элемента и его номера рекомендуется обращение к функции `max`);
2. Написать функцию поиска количества всех отрицательных элементов последнего столбца.

Примечание: поиск суммы элементов ниже главной диагонали осуществляется по формуле

$$sn = \sum_{i=1}^n \sum_{j=1}^n a_{ij}, \quad \text{где } i > j.$$

Поиск суммы элементов выше главной диагонали осуществляется по формуле:

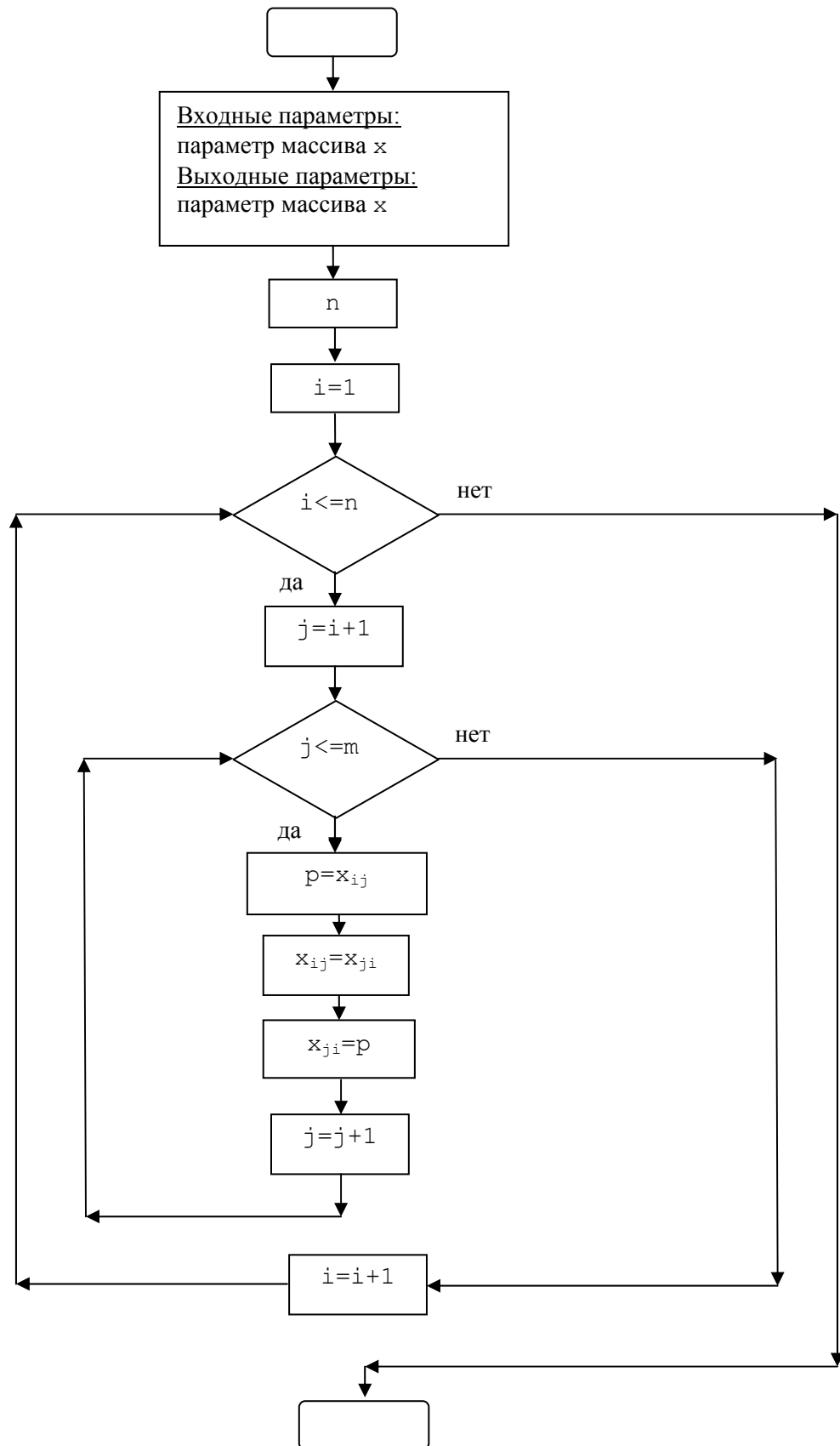
$$sv = \sum_{i=1}^n \sum_{j=1}^n a_{ij}, \quad \text{где } i < j.$$

Задача 8. Даны две квадратные матрицы A и B .

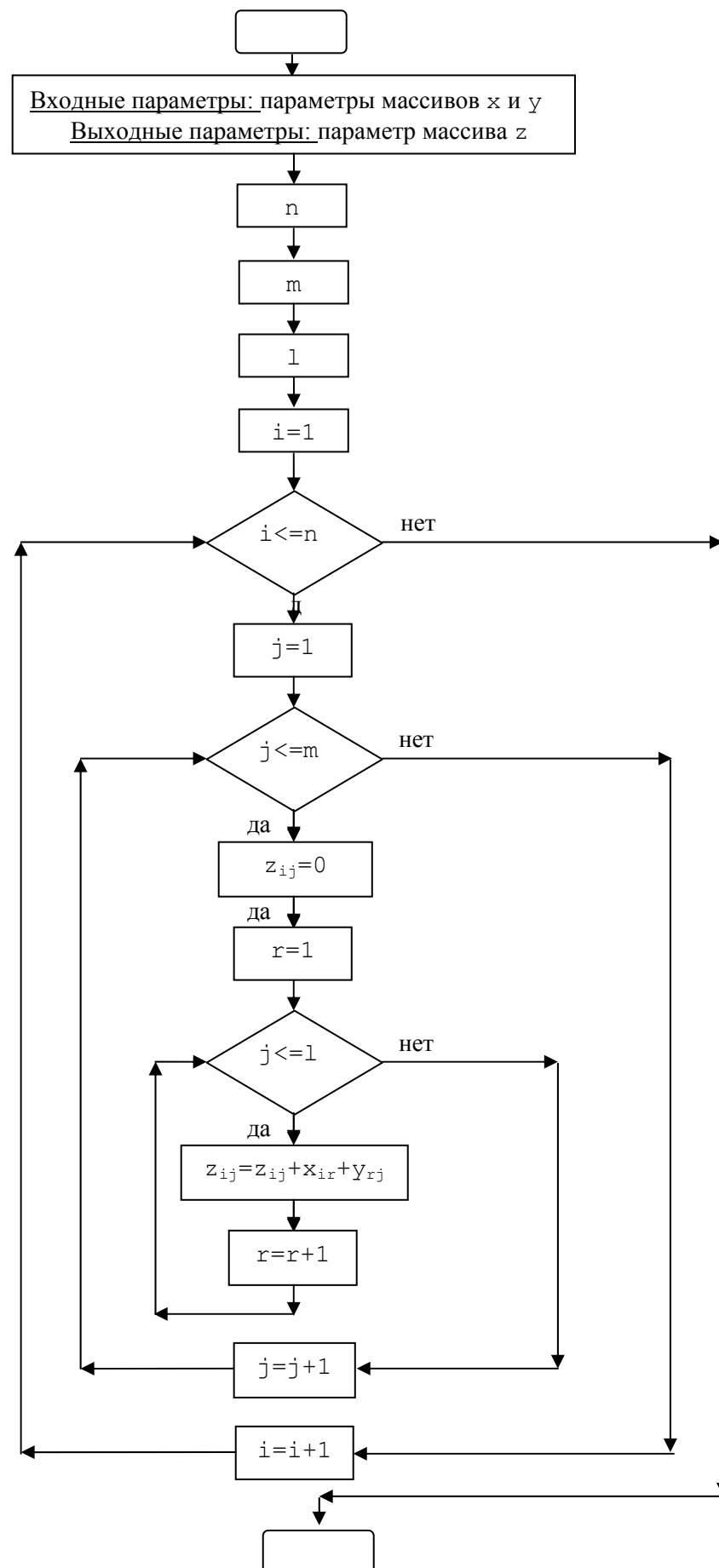
1. Написать функцию транспонирования матрицы. Вывести на экран транспонированные матрицы A и B . Сравнить результат работы функции транспонирования матрицы с результатом работы стандартного действия транспонирования в Scilab `.
2. Написать функцию поиска произведения двух матриц. На примере матриц A и B сравнить результат работы этой функции с результатом работы стандартного действия в Scilab – матричное умножение `\*`.

Блок-схема алгоритма

1. Блок-схема функции транспонирования матрицы.



2. Блок-схема функции поиска произведения двух матриц.



Листинг программного кода

1.

```
function x=trans(x)
    n=size(x,1);
    for i=1:n
        for j=i+1:n
            p=x(i,j);
            x(i,j)=x(j,i);
            x(j,i)=p;
        end
    end
endfunction
```

2.

```
function z=proizv_matr(x, y)
    n=size(x,1);
    m=size(y,2);
    l=size(x,2);
    for i=1:n
        for j=1:m
            z(i,j)=0;
            for r=1:l
                z(i,j)=z(i,j)+x(i,r)*y(r,j);
            end
        end
    end
endfunction
```

Результат работы программы

1.

```
-->exec('C:\Documents and Settings\Мои
документы\work_Scilab\trans.sci', -1)
```

```
-->a=[1 3 2;5 4 3;6 5 4];b=[-5 6 4;-3 4 -3;9 8 0];
-->a
a =
    1.    3.    2.
    5.    4.    3.
    6.    5.    4.
-->b
b =
   -5.    6.    4.
   -3.    4.   -3.
    9.    8.    0.
-->atr=trans(a)
```

```
atr =
    1.    5.    6.
    3.    4.    5.
    2.    3.    4.
```

```
-->btr=trans(b)
```

```
btr =
    5. - 3.    9.
    6.    4.    8.
    4. - 3.    0.
```

```
-->a'
```

```
ans =
    1.    5.    6.
    3.    4.    5.
    2.    3.    4.
```

```
-->b'
```

```
ans =
    - 5. - 3.    9.
    6.    4.    8.
    4. - 3.    0.
```

2.

```
-->exec('C:\Documents and Settings\Мои
документы\work_Scilab\proizv_matr.sci', -1)
```

```
-->a=[1 3 2;5 4 3;6 5 4];b=[-5 6 4;-3 4 -3;9 8 0];
```

```
-->a
```

```
a =
    1.    3.    2.
    5.    4.    3.
    6.    5.    4.
```

```
-->b
```

```
b =
    - 5.    6.    4.
    - 3.    4. - 3.
    9.    8.    0.
```

```
-->pr=proizv_matr(a,b)
```

```
pr =
    4.    34. - 5.
    - 10.   70.    8.
    - 9.    88.    9.
```

```
-->a*b
```

```
ans =
    4.    34. - 5.
    - 10.   70.    8.
    - 9.    88.    9.
```

Задача 9. Дана квадратная матрица A .

1. Найти определитель матрицы A . Если он не равен 0, найти матрицу обратную к A .

2. Написать функцию нахождения суммы всех элементов матрицы. Сравнить результат работы этой функции с результатом работы встроенной в Scilab функции `sum`.

3. Написать функцию удаления всех нечетных столбцов матрицы.

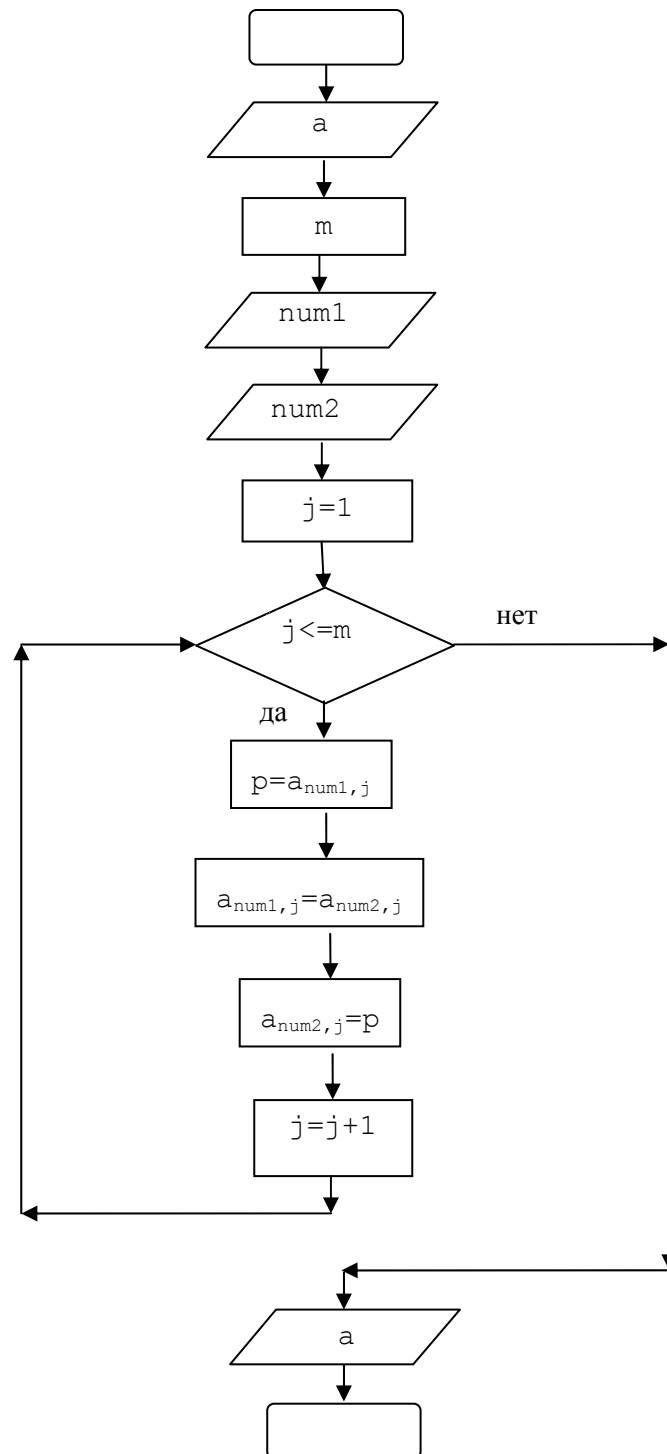
Задача 10. Дана матрица A .

1. Переставить две строки матрицы. Номера строк вводятся с клавиатуры.

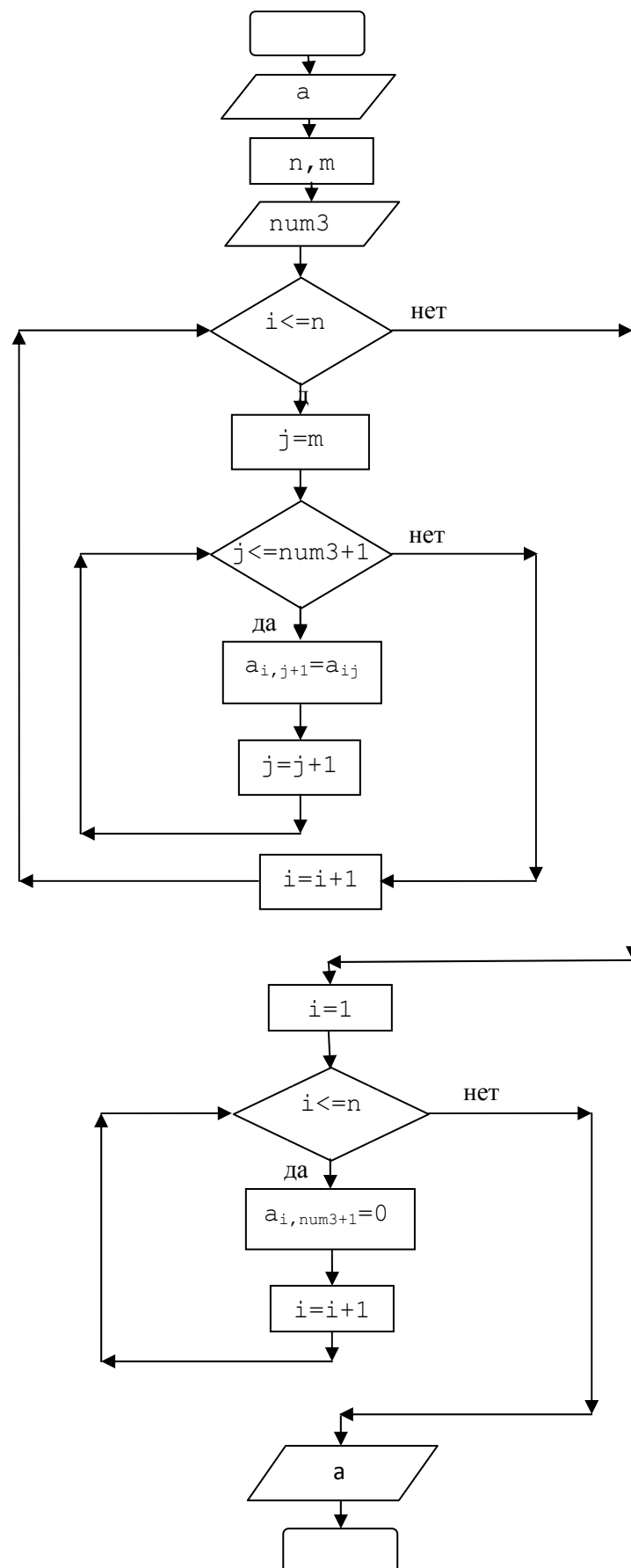
2. Вставить столбец из нулей в матрицу после столбца, номер которого введен с клавиатуры.

Блок-схема алгоритма

1. Блок-схема перестановки двух строк матрицы.



2. Блок-схема вставки столбца из нулей в матрицу.



Листинг программного кода

```
a=[1 3 2;-5 6 7;3 5 2]; //Вводим матрицу
disp(a, 'a=')
n=size(a,1); //Определяем кол-вр строк матрицы a
m=size(a,2); //Определяем кол-во столбцов матрицы a

//Перестановка строк
num1=input('num1='); //Вводим номер первой строки для
перестановки
num2=input('num2='); //Вводим номер второй строки для
перестановки
for j=1:m //Перебираем все столбцы
    //Переставляем строки с номерами num1 и num2
    p=a(num1,j);
    a(num1,j)=a(num2,j);
    a(num2,j)=p
end
disp(a, 'anew=') //Выводим новый массив

//Вставка столбца из нулей
num3=input('num3=') //Номер столбца, после которого
вставка
for i=1:n //Перебираем все строки
    for j=m:-1:num3+1 //Перебираем все столбцы с
последнего до столбца с номером num+1
        a(i,j+1)=a(i,j) //Сдвигаем все столбцы с номером
меньше, чем num+1, влево
    end
end
//Вставляем столбец из нулей на позицию num+1
for i=1:n
    a(i,num3+1)=0
end
disp(a, 'anew2=') //Выводим новый массив
```

Результат работы программы

```
-->exec('C:\Documents and Settings\Мои
документы\work_Scilab\zad18_3.sce', -1)
a=
    1.    3.    2.
   - 5.    6.    7.
    3.    5.    2.
num1=1
num2=3
anew=
    3.    5.    2.
   - 5.    6.    7.
    1.    3.    2.
```

```

num3=2
anew2=
    3.    5.    0.    2.
-   5.    6.    0.    7.
    1.    3.    0.    2.

```

Задача 11. Дана матрица A .

1. Переставить первый столбец матрицы и столбец с минимальным элементом.
2. Вставить строку из нулей перед строкой, в которой содержится максимальный элемент.

Контрольные вопросы

1. Что такое двумерный массив? Как в Scilab задать двумерный массив?
2. Как определить количество строк в двумерном массиве с помощью встроенной в Scilab функции? Количество столбцов?
3. Как в Scilab строки матрицы отделяются друг от друга?
4. Какие методы решения линейных алгебраических уравнений Вы знаете?
5. Какая функция нахождения определителя существует в Scilab?
6. Пусть дана матрица a . Что означает запись в Scilab `inv(a)`? a' ?
7. Как в Scilab удалить m -ю строку в матрице?
8. Пусть дана матрица a

```

a=
    1.    2.    4.    0.
    6.    5.    0.    2.
    1.    2.    3.    5.
    0.    7.    6.    9.

```

Чему будет равна переменная s в результате работы следующей программы в Scilab? Что вычисляется в данном коде?

```

s=0
for i=1:4
    for j=1:4
        if i==j then
            s=s+a(i,j)
        end
    end
end
disp(s)

```

Как изменится результат работы программы, если поставить условие $i < j$ вместо $i == j$?

7. ПРОГРАММИРОВАНИЕ В SCILAB

ИТОГОВАЯ РАБОТА

Цель работы: итоговый контроль за успеваемостью студента.

Постановка задачи: решить приведенные далее задачи.

Задачи

Задача 1. Вычислить сопротивление цепи R , состоящей из резисторов R_1 и R_2 , соединенных:

– последовательно (1 вариант):

– параллельно (2 вариант):

Решить задачу при $R_1 = 20$, $R_2 = 30$.

Задача 2. Составить программу для вычисления суммы S и функции y на заданном интервале изменения x .

1	$S = \frac{x^2}{2} + \frac{x^4}{12} + \dots + (-1)^{n+1} \frac{x^{2n}}{2n(2n-1)}$	$0,1 \leq x \leq 0,8$	10	$y = x \cdot \operatorname{arctg}(x) - \ln \sqrt{1+x^2}$
2	$S = \cos x + \frac{\cos 3x}{3^2} + \dots + \frac{\cos(2n-1)x}{(2n-1)^2}$	$\frac{\pi}{5} \leq x \leq \pi$	40	$y = \frac{\pi^2}{8} - \frac{\pi}{4} x $
3	$S = 3x + 8x^2 + \dots + n(n+2)x^n$	$0,1 \leq x \leq 0,8$	40	$y = \frac{x(3-x)}{(1-x)^3}$
4	$S = x \cos \frac{\pi}{4} + x^2 \cos 2 \frac{\pi}{4} + \dots + x^n \cos n \frac{\pi}{4}$	$0,1 \leq x \leq 0,8$	40	$y = \frac{x \cos(\frac{\pi}{4}) - x^2}{1 - 2x \cos(\frac{\pi}{4}) + x^2}$
5	$S = \frac{x}{3!} + \frac{4x^2}{5!} + \dots + \frac{n^2 x^n}{(2n+1)!}$	$0,1 \leq x \leq 0,8$	20	$y = \frac{1}{4} \left(\frac{x+1}{\sqrt{x}} \cdot \operatorname{sh}(\sqrt{x}) - \operatorname{ch}(\sqrt{x}) \right)$
6	$S = x + \frac{x^3}{3!} + \dots + \frac{x^{2n+1}}{2n(2n+1)}$	$0,1 \leq x \leq 1$	20	$y = \frac{e^x - e^{-x}}{2}$
7	$S = -(1+x)^2 + \frac{(1+x)^4}{2} + \dots + (-1)^n \frac{(1+x)^{2n}}{n}$	$-2 \leq x \leq 0$	40	$y = \ln \frac{1}{2+2x+x^2}$
8	$S = \frac{(2x)^2}{2} + \frac{(2x)^4}{24} + \dots + (-1)^n \frac{(2x)^{2n}}{2n!}$	$0,1 \leq x \leq 1$	15	$y = 2(\cos^2 x) - 1$
9	$S = 1 + \frac{x^4}{12} + \dots + (-1)^{n+1} \frac{x^{2n}}{2n(2n-1)}$	$0,1 \leq x \leq 1$	35	$y = (1-x^2) \cdot \cos(x) - \frac{x}{2} \sin(x)$
10	$S = x - \frac{x^3}{3} + \dots + (-1)^n \frac{x^{2n+1}}{2n+1}$	$0,1 \leq x \leq 0,5$	40	$y = \operatorname{arctg}(x)$

11	$S = 1 + 2\frac{x}{2} + \dots + \frac{n^2 + 1}{n!} \left(\frac{x}{2}\right)^2$	$0,1 \leq x \leq 1$	30	$y = \left(\frac{x^2}{4} + \frac{x}{2} + 1\right)$
12	$S = 1 + \frac{2x}{1!} + \dots + \frac{(2x)^n}{n!}$	$0,1 \leq x \leq 1$	20	$y = e^{2x}$
13	$S = \frac{\cos 2x}{3} + \frac{\cos 4x}{15} + \dots + \frac{\cos 2nx}{4n^2 - 1}$	$0,1 \leq x \leq 0,8$	50	$y = \frac{1}{2} - \frac{\pi}{4} \sin(x) $
14	$S = 1 + \frac{x^2}{2!} + \dots + \frac{x^{2n}}{(2n)!}$	$0,1 \leq x \leq 1$	10	$y = \frac{e^x + e^{-x}}{2}$
15	$S = \sin x + \frac{\sin 3x}{3} + \dots + \frac{\sin(2n-1)x}{2n-1}$	$\frac{\pi}{10} \leq x \leq \frac{9\pi}{10}$	40	$y = \frac{\pi}{4}$
16	$S = \frac{x^3}{3} + \frac{x^5}{15} + \dots + (-1)^{n+1} \frac{x^{2n+1}}{4n^2 - 1}$	$0,1 \leq x \leq 1$	30	$y = \frac{1+x^2}{2} \cdot \operatorname{arctg}(x) - \frac{x}{2}$
17	$S = -\cos x + \frac{\cos 2x}{2^2} + \dots + (-1)^n \frac{\cos nx}{n^2}$	$\frac{\pi}{5} \leq x \leq \pi$	20	$y = \frac{1}{4} \left(x^2 - \frac{\pi^2}{3}\right)$
18	$S = \frac{x-1}{x+1} + \frac{1}{3} \left(\frac{x-1}{x+1}\right)^3 + \dots + \frac{1}{2n+1} \left(\frac{x-1}{x+1}\right)^{2n}$	$0,2 \leq x \leq 1$	10	$y = \frac{1}{2} \ln x$
19	$S = \frac{x \cos \frac{\pi}{3}}{1} + \frac{x^2 \cos \left(2 \frac{\pi}{3}\right)}{12} + \dots + (-1)^{n+1}$	$0,1 \leq x \leq 0,8$	35	$y = -\frac{1}{2} \ln(1 - 2x \cos \frac{\pi}{3} + x^2)$
20	$S = 1 + \frac{\cos x}{1!} + \dots + \frac{\cos nx}{n!}$	$0,1 \leq x \leq 1$	20	$y = e^{\cos x} \cos(\sin x)$

Задача 3. В одномерном массиве, состоящем из n целых элементов:

1. Переставить местами половинки массива.
2. Вставить 0 перед всеми отрицательными элементами массива.
3. Удалить из массива все элементы, в которых последняя цифра четная.

Задача 4. В одномерном массиве, состоящем из n целых элементов:

1. Вставить по одному элементу со значениями, вводимыми с клавиатуры, перед всеми элементами, которые положительны и стоят на четной позиции.
2. Упорядочить массив таким образом, чтобы сначала располагались положительные элементы массива, а потом все остальные в той последовательности, в которой они были до этого.

3. Найти произведение элементов, расположенных между максимальным и минимальным по модулю элементами (для поиска максимального и минимального элементов рекомендуется обращение к функциям `max` и `min`).

Задача 5. Дана квадратная матрица A :

1. Определить минимальное по модулю число на главной диагонали и номер его строки.
2. Определить произведение элементов, имеющих нечетную сумму индексов.
3. На место последнего элемента каждой строки записать сумму предыдущих ее элементов.

Задача 6. Дана квадратная матрица A :

1. Элементы, стоящие на главной диагонали расположить в порядке возрастания, оставив их на главной диагонали.
2. Удалить отрицательные элементы матрицы.
3. Определить среднее арифметическое каждой диагонали матрицы.

Контрольные вопросы

1. Что такое Scilab и каковы его возможности?
2. Какие способы записи вещественных чисел поддерживает Scilab? Как задаются мнимые (комплексные числа)? Какие стандартные константы предопределены в Scilab?
3. Как в Scilab задается многочлен? Как действия над ним определены в Scilab? Как найти корни многочлена?
4. Что такое условный оператор? Как он описывается в Scilab? Опишите принцип работы.
5. Какие операторы цикла Вы знаете? Опишите принцип их работы, нарисуйте блок-схему. Как они задаются в Scilab?
6. Что такое вектор? Что такое матрица? Как они задаются в Scilab? Как обратиться к элементу вектора или матрицы? Какие действия над векторами и матрицами определены в Scilab?
7. Какие способы создания функций в Scilab Вы знаете? Куда записывается результат работы программы-функции? Куда записываются входные аргументы? Как обратиться к функции? Что такое глобальные и локальные переменные?

СПИСОК ЛИТЕРАТУРЫ

1. **Campbell Stephen L.** Modeling and Simulation in Scilab/Scicos / Stephen L. Campbell, Jean-Philippe Chancelier, Ramine Nikoukhah. – Berlin: Springer, 2006. – 313 с.
2. Introduction to Scilab. User's Guide. Scilab Group Rocquencourt, France: INRIA, 2010. <http://www.scilab.org/content/download/247/1702/file/introscilab.pdf>.
3. **Алексеев, Е. Р.** Scilab: Решение инженерных и математических задач / Е. Р. Алексеев, О. В. Чеснокова, Е. А. Рудченко. – М.: ALT Linux; БИНОМ. Лаборатория знаний, 2008. – 260 с.
4. **Андриевский, А.Б.** Использование системы Scilab: практическое пособие / А.Б. Андриевский, Б.Р. Андриевский, А.Л. Фрадков; Балт. гос. техн. ун-т. – СПб., 2010. – 77 с.
5. **Андриевский, Б. Р.** Элементы математического моделирования в программных средах MATLAB5 и Scilab / Б. Р. Андриевский, А. Л. Фрадков – СПб.: Наука, 2001. – 286 с.
6. Сборник задач по программированию на языке Турбо Паскаль 7.0: учеб. пособие. Часть 1 / В. П. Пучков, А. В. Троицкий, А. Б. Лазарева, Д. М. Фомин; НГТУ. – Н. Новгород, 2000. – 54 с.
7. **Тропин, И. С.** Численные и технические расчеты в среде Scilab (ПО для решения задач численных и технических вычислений): учеб. пособие / И. С. Тропин, О. И. Михайлова, А. В. Михайлов. – М.: 2008. – 65 с.
8. Тригонометрическая формула Виета. Википедия [Электронный ресурс] // URL: http://ru.wikipedia.org/wiki/Тригонометрическая_формула_Виета (дата обращения: 25.04.2015).

ПРИЛОЖЕНИЯ

Приложение А

Установка Scilab на ПК

Последнюю версию пакета Scilab всегда можно скачать из сети Internet на странице официального сайта <http://www.scilab.org/>.

Существуют версии Scilab для операционных систем GNU/Linux, Windows XP, Vista, 7, 8, Mac OS X (Intel platforms). Они имеют некоторые отличия в названиях пунктов, но команды пакета во всех версиях идентичны.

Установка пакета для Windows

1. Скачиваем пакет с официального сайта

<http://www.scilab.org/> → Download → Windows

2. Устанавливаем Scilab на ПК

Scilab 5.4.0 Setup → выбрать язык установки (рис. П.1) → OK → Информационное окно (рис. П.2) → Далее → Принять условия лицензионного соглашения (рис. П.3) → Далее → Выбрать путь для установки пакета (оставить выбранный по умолчанию либо изменить с помощью кнопки Обзор...) (рис. П.4) → Далее → Выбрать один из типов инсталляции (инсталляция по умолчанию (Installation Default), полная инсталляция (Full installation) компактная инсталляция (Compact installation), инсталляция с выбором компонентов необходимых для установки (Custom installation) (рис. П.5) → Далее → Выбор расположения ярлыка на компьютере (по умолчанию в меню Пуск либо изменить с помощью кнопки Обзор...) (рис. П.6) → Далее → Выбор дополнительных задач (Create a desktop icon – Создание ярлыка на рабочем столе) (рис. П.8) → Далее → Установить (рис. П.8, П.9) → Далее → Launch Scilab (если этот параметр активен, окно запустится сразу после нажатия кнопки Завершить) → Завершить (рис. П.10).

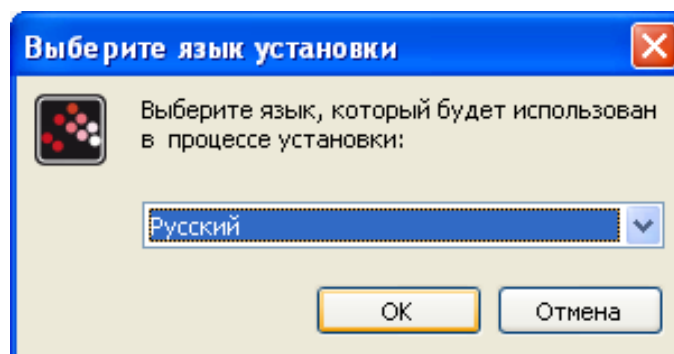


Рис. П.1. Выбор языка установки

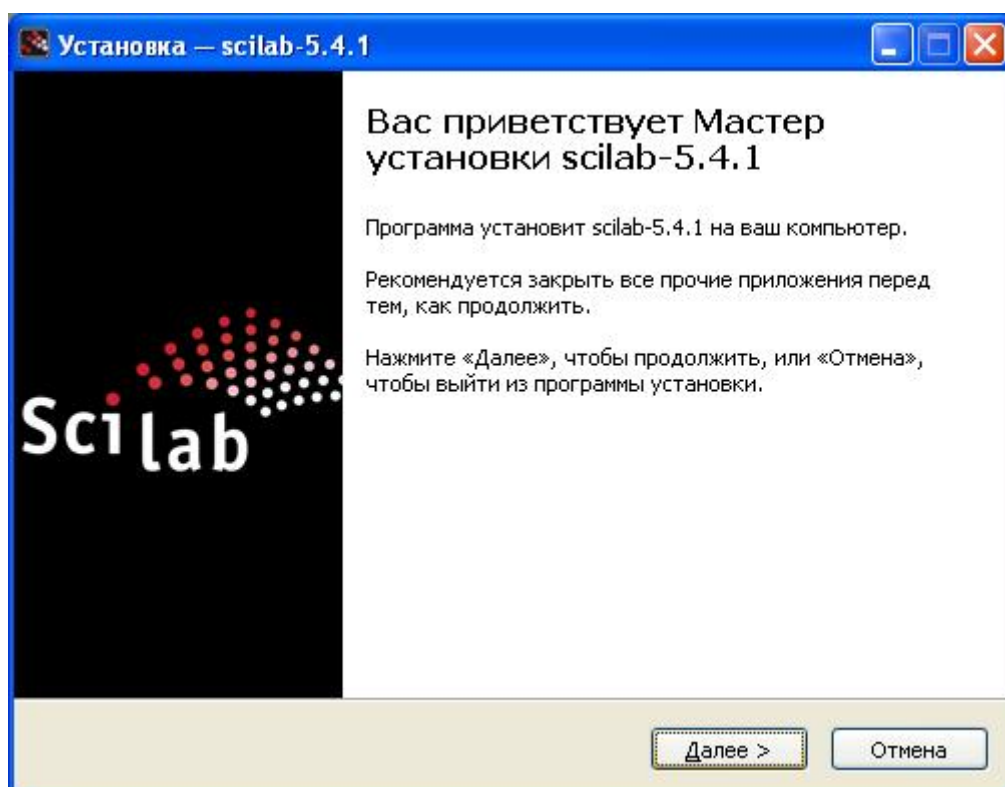


Рис. П.2. Информационное окно

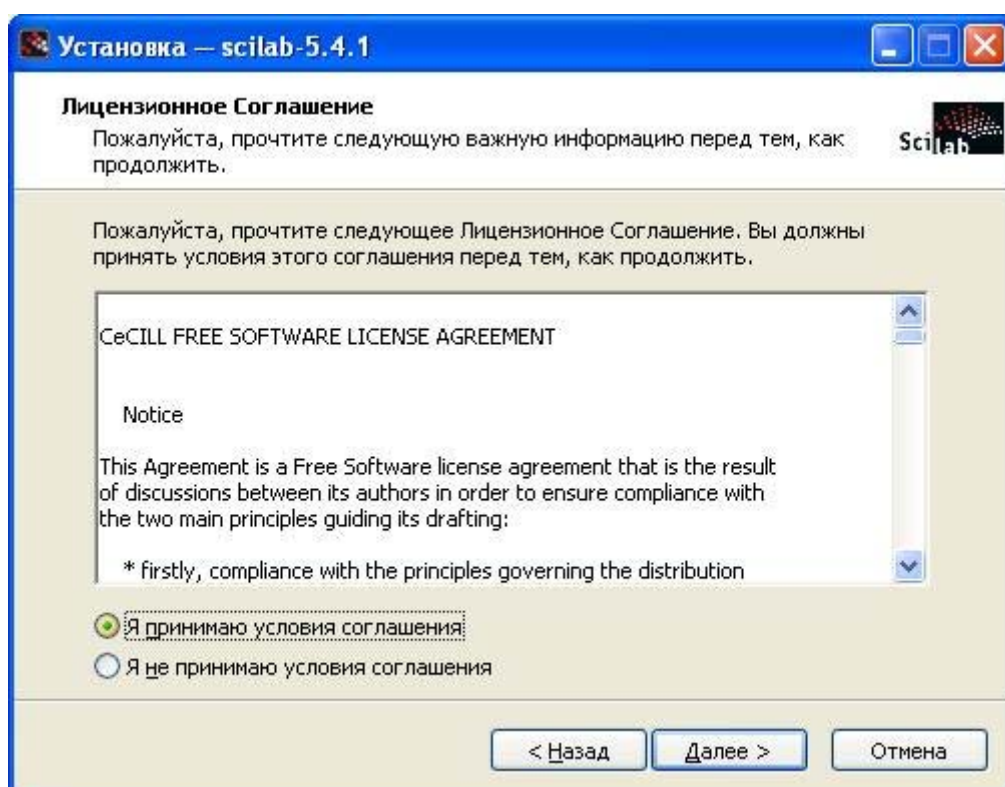


Рис. П.3. Лицензионное соглашение

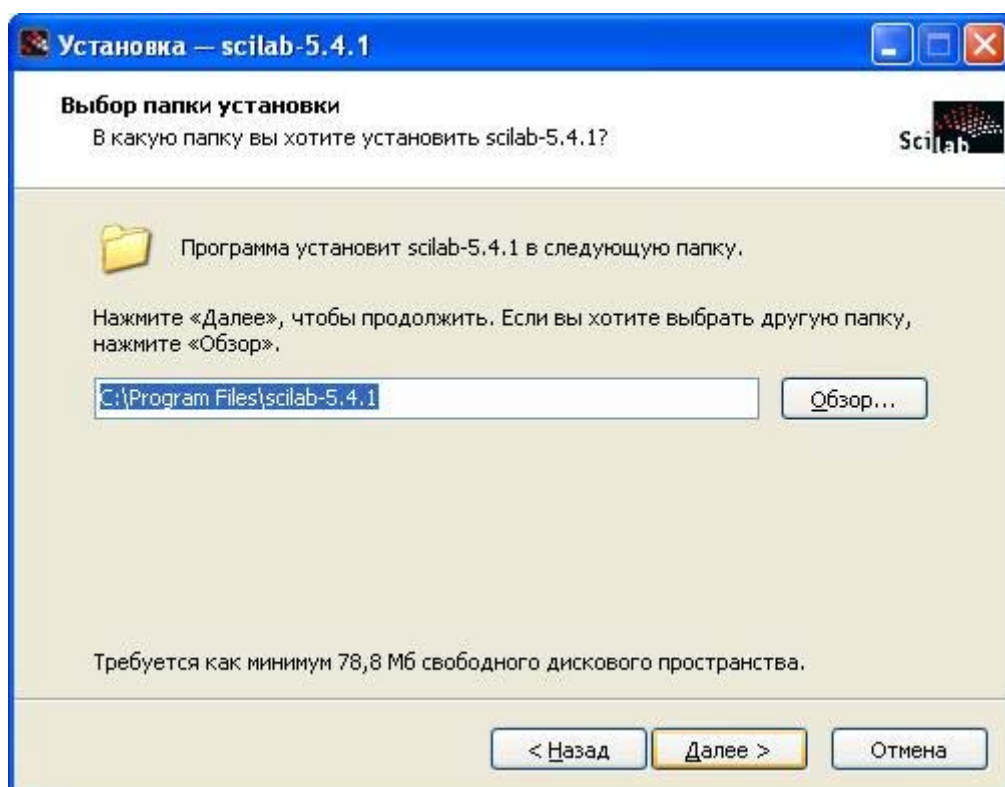


Рис. П.4. Выбор папки установки

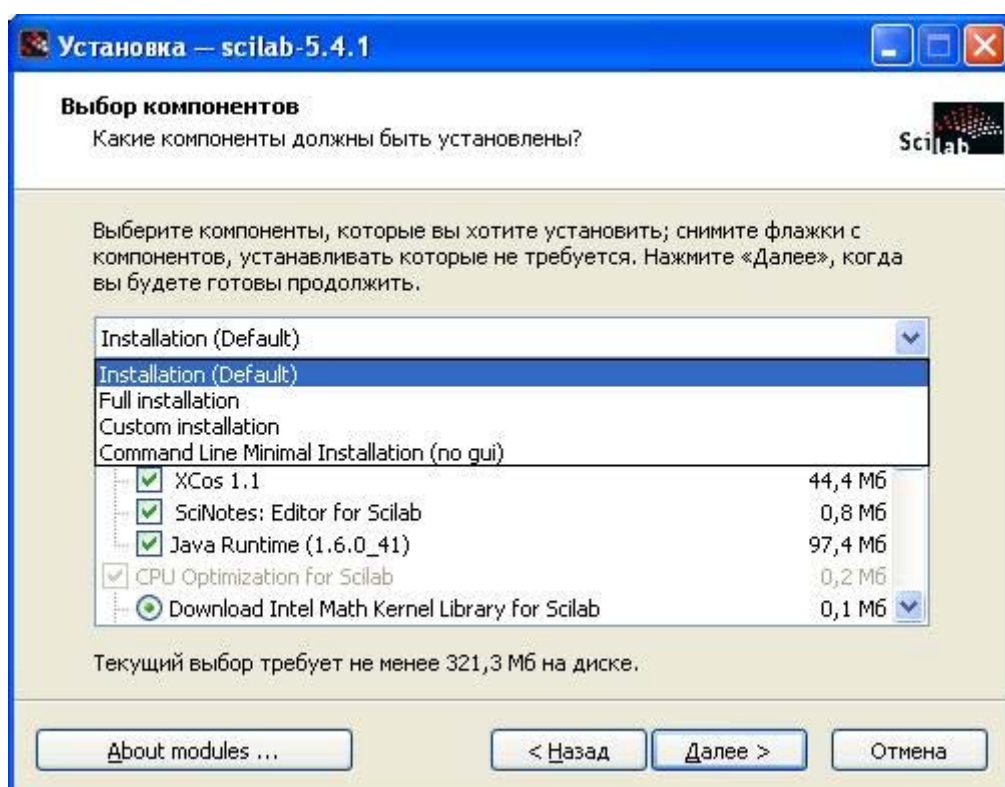


Рис. П.5. Выбор компонентов установки

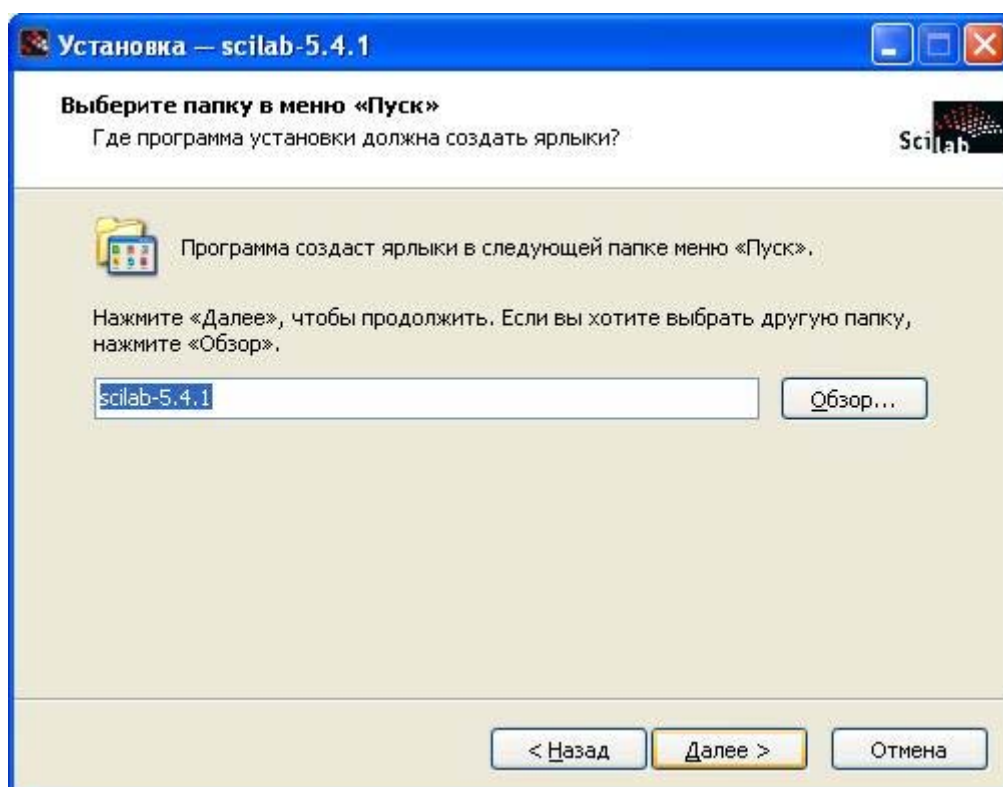


Рис. П.6. Выбор расположения ярлыка на компьютере

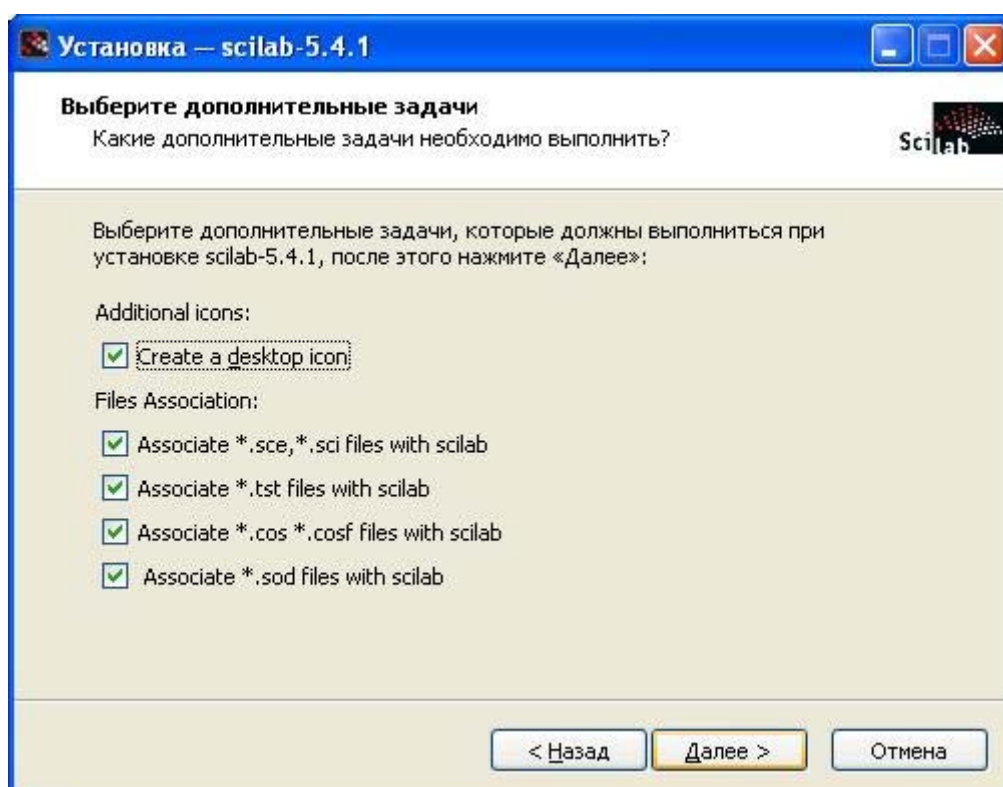


Рис. П.7. Выбор дополнительных задач

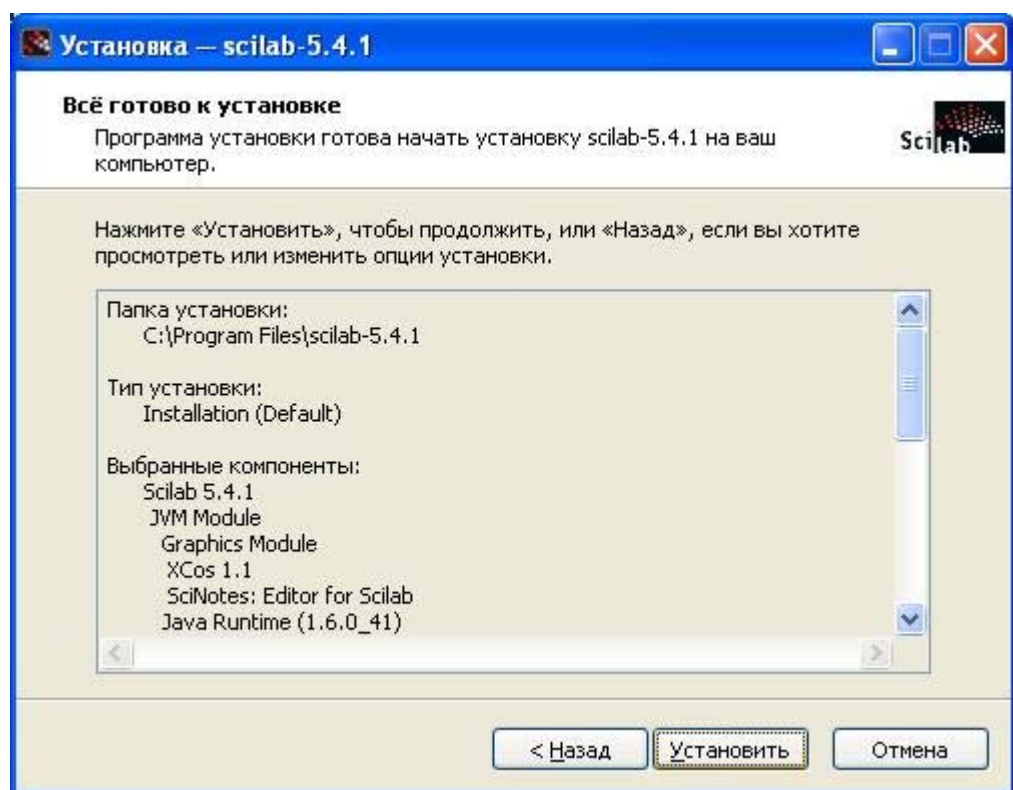


Рис. П.8. Готовность к установке

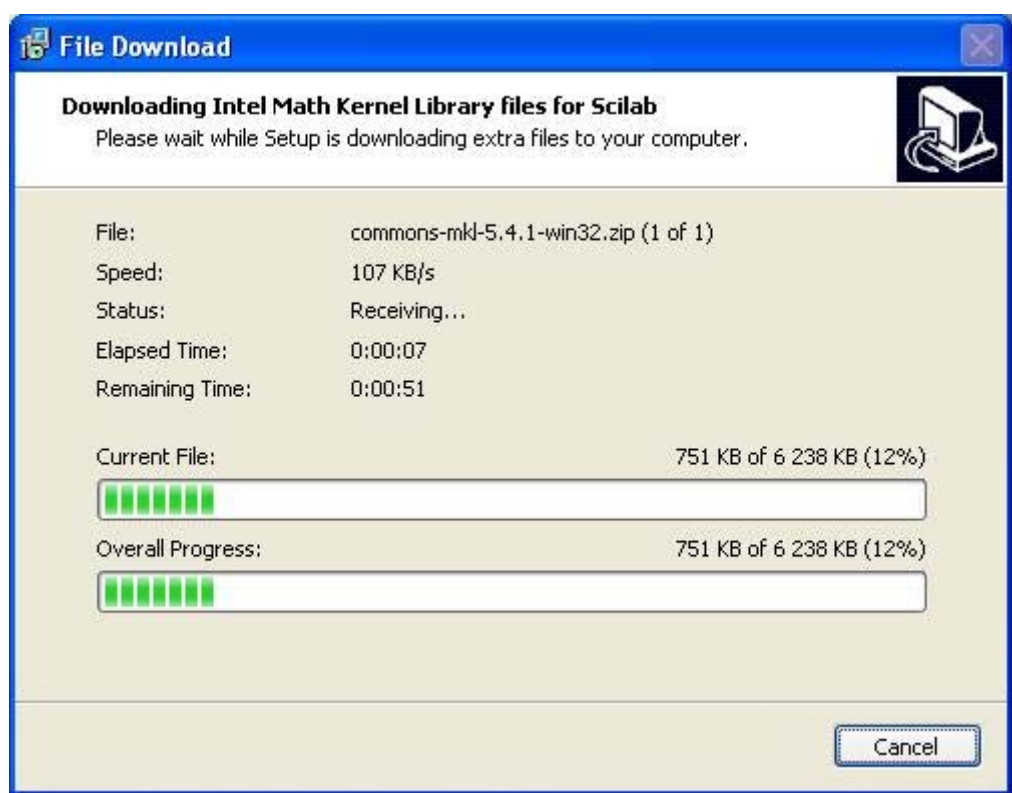


Рис. П.9. Установка

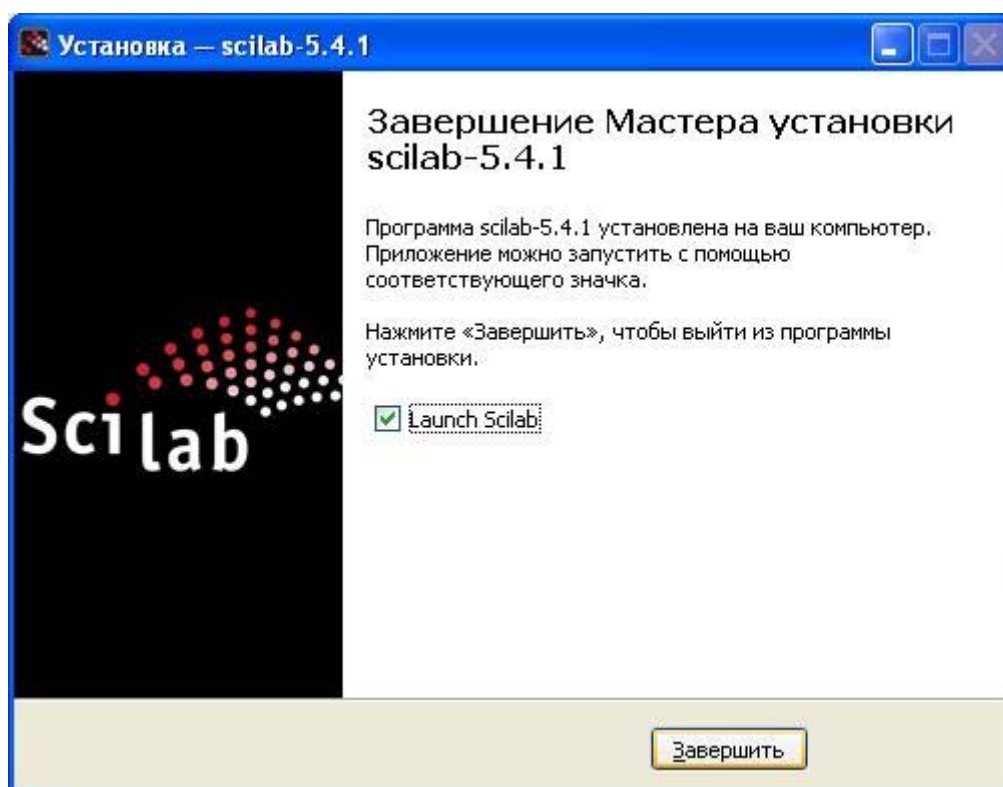
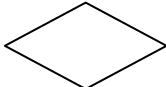


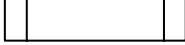
Рис. П.10. Завершение установки

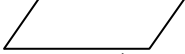
Основные элементы блок-схемы

 - блок начала/конца алгоритма. Начало, конец алгоритма, вход и выход в подпрограмму. Используется обычно с надписями «Начало», «Конец».

 - блок вычислений. Выполнение некоторой операции (арифметической, логической либо иной), в результате которой каким-либо образом изменяются данные. Возможно объединение нескольких операций в один блок. Внутри фигуры записывают непосредственно сами операции.

 - блок условия (логический блок). Выбор одного из двух (или более) возможных решений алгоритма. Внутри элемента расположено условие. Из углов ромба выходят возможные пути, обозначающиеся «да», «нет» (в случае более двух решений обозначения зависят от задачи). Если выходов больше трех, то их следует показывать одной линией, выходящей из вершины (чаще нижней) элемента, которая затем разветвляется.

 - predetermined process. Обращение к подпрограмме. Например, в программировании – вызов процедуры или функции, модуля (части программы вне текущего последовательного кода). Внутри символа записывается название процесса и передаваемые в него данные. Т.е. выполнение процесса, состоящего из одной или нескольких операций, который определен в другом месте программы (в подпрограмме, модуле).

 - ввод-вывод данных. Операции ввода-вывода (чтение, печать и т.д.) в общем виде. Данный блок не определяет носителя данных.

 - соединитель / межстраничный соединитель. Используется для обрыва линии связи в одном месте и продолжения в другом. Внутри элемента блок-схемы вводится уникальный идентификатор.

 - линия потока. Объединяет между собой элементы блок-схемы, указывает последовательность выполнения действий.

 - комментарий. Пометка неактивной части программы. Используется для более подробного описания шага, процесса или группы процессов. Описание помещается со стороны квадратной скобки. Пунктирная линия идет к описываемому элементу, либо группе элементов (при этом группа выделяется замкнутой пунктирной линией).

Образец оформления лабораторной работы

**МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. Р.Е. Алексеева»
АРЗАМАССКИЙ ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ (ФИЛИАЛ)**

**ЛАБОРАТОРНАЯ РАБОТА №1
ПО ИНФОРМАТИКЕ
«ПРОГРАММИРОВАНИЕ ЗАДАЧ ЛИНЕЙНОЙ СТРУКТУРЫ»**

**Выполнил:
студент гр. АСП 2015-1
Иванов И.И.**

**Проверил:
преподаватель
Емельянова Ю.П.**

Арзамас
2015 г.

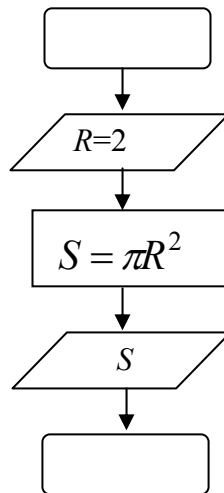
Цель работы: приобретение практических навыков разработки и отладки простейших программ, реализующих линейные вычислительные процессы.

Постановка задачи: решить следующие задачи, имеющие линейную структуру алгоритма решения.

Задача 1. Найти площадь круга, если радиус круга $R = 2$.

Решение задачи: площадь круга вычисляется по формуле $S = \pi R^2$.

Блок-схема алгоритма



Листинг программного кода:

```
-->R=2; S=%pi *R^2
```

Результат работы программы:

```
S =  
12.566371
```

Выводы

- были освоены принципы программирования задач, имеющих линейную структуру алгоритма решения;
- изучены основные принципы работы в Scilab (работа в командном окне среды Scilab, основы работы в редакторе SciNotes, запуск приложения из редактора, операторы ввода/вывода в среде Scilab).

Ответы к задачам для самостоятельного решения

Ответы к лабораторной работе 1

Задача 4 $S = 40$

Задача 6 $\vec{v} = \{3; -1\}$

Ответы к лабораторной работе 2

Задача 3 1) $x_a=1; y_a=1; R=3$ попадает

2) $x_a=2; y_a=5; R=3$ не попадает

Задача 5 Результат совпадает с результатами в задаче 11

1) при $a=4, b=-5$ и $c=1$

ans =

- 1.
- 0.5
1.
0.5

2) при $a=3, b=8$ и $c=-1$

ans =

- 6.830D-17 + 1.6692213i
- 6.830D-17 - 1.6692213i
- 0.3458800
0.3458800

3) при $a=1, b=2$ и $c=3$

ans =

- 0.6050003 + 1.1687709i
- 0.6050003 - 1.1687709i
0.6050003 + 1.1687709i
0.6050003 - 1.1687709i

Задача 6 1) при $a=8, b=7, c=-4, d=1$

1 действительный и пара комплексных корней

$x_3 = 0.2244823 - 0.2098120i$

$x_2 = 0.2244823 + 0.2098120i$

$x_1 = -1.3239647$

2) при $a=1, b=-6, c=11, d=-6$

3 действительных корня

$x_3 = 2.$

$x_2 = 3.$

$x_1 = 1.$

3) при $a=2, b=-11, c=12, d=9$

2 действительных корня

$x_3 = 2.3626543$

$x_2 = 2.3626543$

$x_1 = 0.7746914$

Результаты совпадают с результатами, полученными при помощи функции
roots

Задача 7

4. 4.2415355 - 1.6987739 1.4572384	5. - 2.179509 - 0.4102455 + 1.7445433i - 0.4102455 - 1.7445433i	6. 2.1306624 - 1. 0.4693376	7. - 0.6666667 + 0.4714045i - 0.6666667 - 0.4714045i 0
8. 3. + 5.228D-08i 3. - 5.228D-08i - 0.5	9. 5. - 2. 2.	10. 1. + 1.2247449i 1. - 1.2247449i 0.5	11. - 2.0872797 + 1.1713121i - 2.0872797 - 1.1713121i 0.1745594
12. - 0.5576818 + 1.0425361i - 0.5576818 - 1.0425361i 0.7153636	13. 5. + 7.525D-08i 5. - 7.525D-08i - 1.	14. 5. 1. + 1.785D-08i 1. - 1.785D-08i	15. 6.9101699 - 0.4550849 + 0.2869254i - 0.4550849 - 0.2869254i
16. - 2. 1.0000000 1.	17. - 1.1378411 + 1.5273123i - 1.1378411 - 1.5273123i 0.2756822	18. - 2. 1. 0.6666667	19. - 2.257334 1. 0.5906673
20. 2. 2. 1.	21. 0.5 + 1.3228757i 0.5 - 1.3228757i - 1.	22. - 1.8968053 1.2301386 0	23. - 6. 6. 3.

Задача 9 1996 год крысы

Ответы к лабораторной работе 3

Задача 2 $m=5$ $m!=120$

Задача 6 $n=4$ $chislo=2$ $chislo=3$ $chislo=-1$ $chislo=2$ $p=-12$

Задача 7 $n=5$ $chislo=9$ $chislo=3$ $chislo=5$ $chislo=4$ $chislo=2$
 $k_chet=2$ $k_nechet=3$

Задача 8 $n=9$ $k=2$

Задача 9 $a=2$ $b=3$ $NOK=6$

Результат работы программы совпадает с результатом работы встроенной в Scilab функции `lcm`.

Задача 10 $a=70$ $b=105$ $NOD=35$

Результат работы программы совпадает с результатом работы встроенной в Scilab функции `gcd`.

Задача 11 $x=9834$ $kol-vo$ $cifr=4$

Ответы к лабораторной работе 5

Задача 2 $-->x=[-5 \ 6 \ 3 \ -10 \ 9 \ 8 \ 0 \ 3 \ 4]$

$x = -5. \quad 6. \quad 3. \quad -10. \quad 9. \quad 8. \quad 0. \quad 3.$

4.

$-->[Min, Mini]=minimal(x)$

$Mini = 4.$

$Min = -10.$

Результат работы функции `minimal` совпадает с результатом работы встроенной в Scilab функции `min`.

Задача 4

$a = -9. \quad 5. \quad 6. \quad 12. \quad 3. \quad -4. \quad -2. \quad 6. \quad -6. \quad -$

7.

1)

s= - 28.

2)

pr= 42.

3)

anew1= 3. - 4. - 2. - 9. 5. 6. 12. 6. - 6. - 7.

Задача 6 -->x=[-5 6 3 -10 9 8 0 3 4]

x = - 5. 6. 3. - 10. 9. 8. 0. 3.

4.

1)

-->summa=s_5_3(x)

summa = 9.

2)

-->xnew=udal_nechet(a)

xnew = 6. - 10. 8. 3.

3)

-->xnew1=sort_poslmin(x)

xnew1 = - 10. - 5. 0. 3. 3. 4. 6. 8.

9.

Результат работы программы sort_poslmin совпадает с результатом работы встроенной в Scilab функции mtlb_sort.

Задача 8

a= 9. 7. - 2. 2. 3. - 4. - 2. 6. 6. 7.

1)

el= 10

el= 20

el= 30

el= 40

anew2= 9. 40. 7. - 2. 30. 2. 3. - 4. - 2. 20. 6. 6. 10.

7.

2)

anew1= 9. 7. 2. 3. 6. 6. 7. - 2. - 4. - 2.

3)

pr= - 84.

Ответы к лабораторной работе 6

Задача 5 -->x=[3 2 -6;-4 5 0;-3 4 1]

x =

3. 2. - 6.

- 4. 5. 0.

- 3. 4. 1.

-->[el,numi,numj]=minimaln(x)

numj = 3.

numi = 1.

el = - 6.

Результат работы функции minimaln совпадает с результатом работы встроенной в Scilab функции min.

Задача 7

-->x=[3 2 -6;-4 5 0;-3 4 1]

```

x =
    3.    2.   - 6.
   - 4.    5.    0.
   - 3.    4.    1.

```

1)

```
-->s=diag_vishe(x)
```

```
s = - 4.
```

```
-->s=diag_nizhe(x)
```

```
s = - 3.
```

2)

```
-->xnew=udal_stolbca(x)
```

```
xnew =
```

```
    3.   - 6.
```

```
   - 4.    0.
```

```
   - 3.    1.
```

3)

```
-->k=k_otr_matrix(x)
```

```
k = 1.
```

Задача 9 -->a=[4 3 -5; -4 5 2;-5 0 8]

```
a =
```

```
    4.    3.   - 5.
```

```
   - 4.    5.    2.
```

```
   - 5.    0.    8.
```

1)

```
-->det(a)
```

```
ans = 101.
```

```
-->inv(a)
```

```
ans =
```

```
    0.3960396   - 0.2376238    0.3069307
```

```
    0.2178218    0.0693069    0.1188119
```

```
    0.2475248   - 0.1485149    0.3168317
```

2)

```
-->s=summa(a)
```

```
s = 8.
```

Результат работы функции `summa` совпадает с результатом работы встроенной в Scilab функции `sum`.

3)

```
-->anew=udal_nechet_st(a)
```

```
anew =
```

```
    3.
```

```
    5.
```

```
    0.
```

Задача 11

```
a=
```

```
    1.   - 9.    0.
```

```
    3.    4.   - 3.
```

```
    0.   10.   - 4.
```

1)

anew1=

-	9.	1.	0.
	4.	3.	- 3.
	10.	0.	- 4.

2)

anew2=

1.	-	9.	0.
3.		4.	- 3.
0.		0.	0.
0.		10.	- 4.

Ответы к лабораторной работе 7

Задача 1 1 вариант $R_{pos1} = 50$; 2 вариант $R_{par1} = 12$

**Емельянова Юлия Павловна
Пакшин Павел Владимирович**

ПРОГРАММИРОВАНИЕ В SCILAB

Лабораторный практикум

Редактор Н.Н. Максимова
Технический редактор Т.П. Новикова

Подписано в печать 06.07.2015. Формат 60×84¹/₁₆.
Бумага офсетная. Печать офсетная. Усл. печ. л. 7,25.
Тираж 70 экз. Заказ

Нижегородский государственный технический университет им. Р.Е. Алексеева.

Типография НГТУ.

Адрес университета и полиграфического предприятия:
603950, г. Нижний Новгород, ул. Минина, 24