Министерство науки и высшего образования Российской Федерации

Казанский национальный исследовательский технический университет – КАИ им. А.Н. Туполева

Институт компьютерных технологий и защиты информации Отделение СПО ИКТЗИ «Колледж информационных технологий»

# ТЕОРИЯ АЛГОРИТМОВ

Методические указания к лабораторным работам

Казань 2024

**ОГЛАВЛЕНИЕ**

[ПРОЦЕСС СДАЧИ ВЫПОЛНЕННОЙ РАБОТЫ 2](#_bookmark0)

[ЛАБОРАТОРНАЯ РАБОТА №1 СОСТАВЛЕНИЕ АЛГОРИТМОВ РЕШЕНИЯ ЗАДАЧ НА АЛГОРИТМИЧЕСКОМ ЯЗЫКЕ. 3](#_bookmark1)

**ПРОЦЕСС СДАЧИ ВЫПОЛНЕННОЙ РАБОТЫ**

По итогам выполнения работы студент:

1. Демонстрирует преподавателю отчет о выполненной работе;
2. Демонстрирует приобретённые знания и навыки, отвечает на вопросы преподавателя по выполненному заданию, возможностям её доработки;
3. Демонстрирует отчет по выполненной лабораторной работе.

Итоговая оценка складывается из оценок по трем указанным составляющим.

**ЛАБОРАТОРНАЯ РАБОТА №1**

СОСТАВЛЕНИЕ АЛГОРИТМОВ РЕШЕНИЯ ЗАДАЧ НА АЛГОРИТМИЧЕСКОМ ЯЗЫКЕ.

ЦЕЛЬ

Усвоить понятия: алгоритм, способы описания алгоритва, основные типы алгоритмов, освоить принципы решения задач с использованием основных алгоритмических конструкций.

ХОД РАБОТЫ

1. Псевдокод

**Псевдокод** – это специальный способ описания алгоритма или программы с использованием структурированного языка, который напоминает код программирования, но не привязан к конкретному языку программирования.

**Основные принципы** псевдокода:

1. Первый принцип псевдокода – **читабельность**. Записи должны быть понятны не только вам, но и другим разработчикам. Используйте естественный язык, чтобы формулировать шаги алгоритма.
2. Важным аспектом является **структурированность**. Каждый элемент алгоритма следует четко организовать: использовать отступы для блоков кода, четко выделять начало и конец циклов, условных операторов. Это делает текст легко воспринимаемым и упрощает его анализ.
3. Не менее важным принципом является **конкретность**. Несмотря на абстрактность псевдокода, в нем необходимо точно выражать действия и условия.
4. Кроме того, следует помнить о **последовательности**. Используйте одинаковые обозначения и формулировки для одних и тех же действий на протяжении всего текста. Это предотвращает недоразумения и позволяет сосредоточиться на логике алгоритма.
5. Наконец, актуальность псевдокода лежит в его **простоте**. Избегайте сложных конструкций и терминологии, стремитесь к минимализму и ясности. Это позволит делать описание алгоритмов интуитивно понятным и подготовленным для дальнейшей реализации на различных языках программирования.

Понимание и применение этих ключевых принципов псевдокода существенно упрощает процесс первоначального этапа разработки алгоритмов, что в конечном итоге способствует качественной и быстрой реализации готового решения.

Описание алгоритмов на полупрограммном языке имеет множество неоспоримых **преимуществ**. Опуская конкретную синтаксическую структуру, можно сконцентрироваться на логической части и значимых шагах решения задачи, что существенно упрощает процесс разработки и тестирования:

1. **Ясность и простота**: идея псевдокода заключается в создании понятной и читаемой структуры алгоритма. Даже люди без глубоких знаний в программистских языках, могут легко понять основные шаги решения.
2. **Универсальность**: описание алгоритма на псевдоязыке не привязано к конкретной платформе или синтаксису. Это делает работу более гибкой и удобной при переходе с одного языка программирования на другой.
3. **Эффективное планирование**: перед началом написания кода на языке программирования, важно четко представлять все шаги решения задачи. Псевдокод позволяет сделать это без затраты времени на отладку и исправление синтаксических ошибок.

Псевдокод не требует знания конкретных языков программирования, что делает его идеальным инструментом для начинающих. Он использует простые конструкции такие как 'если', 'пока', 'иначе', которые помогают в организации логических инструкций. Эти конструкции напоминают обычные условия и циклы в программировании, но описываются простым языком.

**'Если'** используется для описания условий. Например, "Если сегодня суббота, то можно спать до обеда".

**'Пока'** применяется для описания циклов. Например, "Пока на улице идет дождь, оставайся дома".

**'Иначе'** предлагает альтернативное действие. Например, "Если завтра будет солнечно, идем на пикник. Иначе останемся дома".

1. Блок-схема

**Блок-схемы** – это способ графического представления алгоритма, в котором шаги изображаются в виде блоков различной формы, соединенных между собой стрелками.

С **основными элементами** блок-схем можно ознакомиться, открыв **ГОСТ 19.701**.

1. Пример выполнения лабораторной работы

**Задание 1.** Определить площадь трапеции по введенным значениям оснований (a и b) и высоты (h).

Запись решения задачи на алгоритмическом языке:

|  |
| --- |
| алг трапеция  вещ a,b,h,s  нач  ввод f,b,h   s:=((a+b)/2) \* h   вывод s  кон |

Запись алгоритма в виде блок-схемы (Рисунок 1):

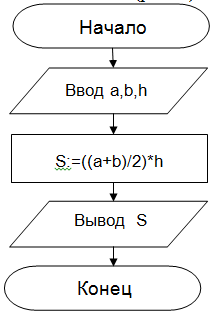


Рисунок 1 – Блок-схема линейного алгоритма

|  |
| --- |
| float a = float.Parse(Console.ReadLine());  float b = float.Parse(Console.ReadLine());  float h = float.Parse(Console.ReadLine());  float S;  S = ((a + b) / 2) \* h;  Console.WriteLine(S); |

**Задание 2.** Определить среднее арифметическое двух чисел, если a положительное и частное (a/b) в противном случае.

Запись решения задачи на алгоритмическом языке:

|  |
| --- |
| алг числа      вещ a,b,c  нач      ввод a,b      если a>0          то    с:=(a+b)/2          иначе с:=a/b      все      вывод с  кон |

Запись алгоритма в виде блок-схемы (Рисунок 2):

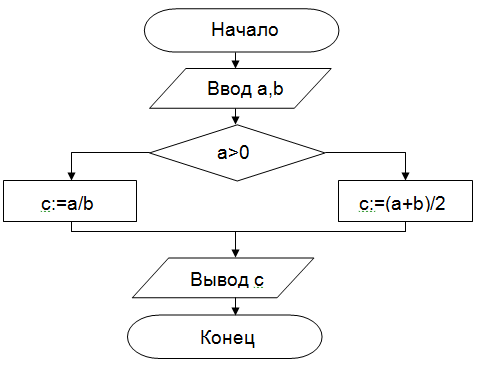


Рисунок 2 – Блок-схема алгоритма с ветвлением

|  |
| --- |
| float a = float.Parse(Console.ReadLine());  float b = float.Parse(Console.ReadLine());  float C;  if (a > 0) C = (a + b) / 2;  else C = a / b;  Console.WriteLine(C); |

**Задание 3.** Составить алгоритм нахождения суммы целых чисел в диапазоне от 1 до 10.

Запись решения задачи на алгоритмическом языке:

|  |
| --- |
| алг сумма      вещ a,s  нач      S:=0;      а:=1;      нц          пока a<=10          S:=S+a;          а:=a+1;      кц      вывод S  кон |

Запись алгоритма в виде блок-схемы (Рисунок 3):

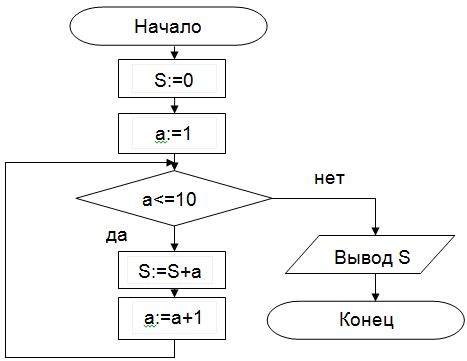


 Рисунок 3 – Циклический алгоритм с предусловием

|  |
| --- |
| float S = 0;  float а = 1;  while (A <= 10)  {  S += а;  а++;  }  Console.WriteLine(S); |

1. Варианты заданий

**Задание 1.**  Составить алгоритм решения задачи с помощью алгоритмического языка псевдокод и с помощью блок-схем, используя конструкцию линейного алгоритма. Реализовать на любом языке программирования.

1. Вычислить площадь поверхности и объем усеченного конуса по следующим формулам:

2. Вычислить координаты центра тяжести трех материальных точек с массами *m1, m2, m3* и координатами *(x1, y1), (x2, y2), (x3, y3)*   по формулам:

3.Вычислить площадь треугольника со сторонами *a, b, c* по формуле Герона:

где p – полупериметр, вычисляемый по формуле:

4. Вычислить координаты точки, делящей отрезок *а1а2 в* отношении *n1: n2* по формулам:

где

1. Вычислить медианы треугольника со сторонами *a, b, c* по формулам:

6. Вычислить площадь круга и длину окружности по введенному значению радиуса.

7. Вычислить площадь *S* и периметр *L* эллипса по введенным значениям полуосей *a* и *b*:

8. Вычислить объем *V* и площадь боковой поверхности цилиндра *S* по введенным значениям радиуса основания *R* и высоты цилиндра *H*.

9.  Вычислить объем *V* и площадь боковой поверхности конуса *S* по введенным значениям радиуса основания *r*, высоты *h* и образующей l:

10.  Вычислить объем *V* и площадь поверхности *S* сферы по введенному значению радиуса *r*:

11. Дано целое четырехзначное число. Используя операции *div* и *mod*, найти сумму его цифр.

12. Дана сторона равностороннего треугольника. Найти площадь этого треугольника и радиусы вписанной и описанной окружностей.

13. Даны координаты трех вершин треугольника *(x1 , y1), (x2 , y2),  (x3 , y3).* Найти его периметр и площадь.

14. Дана длина окружности. Найти площадь круга, ограниченного этой окружностью.

15. Дана площадь круга. Найти длину окружности, ограничивающей этот круг.

**Задание 2.** Составить алгоритм решения задачи с помощью алгоритмического языка псевдокод и с помощью блок-схем, используя конструкцию алгоритма с ветвлением. Реализовать на любом языке программирования.

1. Составить программу для решения квадратного уравнения:

2. Определить максимальное четное число из двух введенных.

3. Определить, можно ли из отрезков с длинами *х*, *y* и *z* построить треугольник.

4. Ввести два числа *а* и *b*. Большее число заменить утроенным произведением, меньшее – полусуммой.

5. Если среди трех чисел *a*, *b*, *c* имеется хотя бы одно четное, то найти максимальное число, иначе – минимальное.

6. Определить, в каком квадранте находится точка с координатами *x* и *y* и вывести номер квадранта на экран.

7. Найти квадрат наибольшего из двух чисел *а* и *b*. Вывести на экран число 1, если наибольшим является число *а*, число 2 – если наибольшим числом является *b*.

8. Определить, попадает ли точка с координатами *x* и *y* в круг радиусом *R*. Если точка попадает в круг, вывести на экран единицу, в противном случае – ноль.

9. Написать алгоритм решения задачи, которая решает уравнение *ax + b = 0* относительно *x* для любых чисел *a* и *b*, введенных с клавиатуры. Все числа считаются действительными.

10. Написать алгоритм решения задачи, которая определяет, лежит ли точка *А(х,у)* внутри некоторого кольца («внутри» понимается в строгом смысле, т.е. случай, когда точка *А* лежит на границе кольца, недопустим). Центр кольца находится в начале координат. Для кольца заданы внутренний и внешний радиусы *r1, r2*. Координаты *x* и *у* вводятся с клавиатуры.

11. Даны две переменные целого типа: *A* и *B*. Если их значения не равны, то присвоить каждой переменной произведение этих значений, а если равны, то присвоить переменным нулевые значения.

12. Даны две переменные целого типа: *A* и *B*. Если их значения не равны, то присвоить каждой переменной минимальное из этих значений, а если равны, то присвоить переменным нулевые значения.

13. Даны целочисленные координаты точки на плоскости. Если точка не лежит на координатных осях, то вывести 0. Если точка совпадает с началом координат, то вывести 1. Если точка не совпадает с началом координат, но лежит на оси *OX* или *OY*, то вывести соответственно 2 или 3.

14. Даны вещественные координаты точки, не лежащей на координатных осях *OX* и *OY*. Вывести номер координатной четверти, в которой находится данная точка.

15. Дано целое число, лежащее в диапазоне от –999 до 999. Вывести строку – словесное описание данного числа вида «отрицательное двузначное число», «нулевое число», «положительное однозначное число» и т.д.

**Задание 3.** Составить алгоритм решения задачи с помощью алгоритмического языка псевдокод и с помощью блок-схем, используя конструкцию циклического алгоритма. Реализовать на любом языке программирования.

1.Найти сумму чисел, кратных трем, в диапазоне от 0 до 50.

1. Найти сумму первых десяти чисел, кратных пяти.
2. Найти произведение четных чисел в диапазоне от 2 до 30.
3. Вводятся положительные числа. Прекратить ввод, когда сумма введенных чисел превысит 100.
4. Требуется найти сумму чисел, кратных 7, в диапазоне от 0 до 100. Вывести на экран сумму чисел и их количество.
5. Определить количество целых чисел, кратных 3 (от 3 и далее), дающих в сумме число, превышающее 200.
6. Вводятся 10 чисел. Вывести на экран суммы положительных и отрицательных чисел и их количество.
7. Вывести на экран значения функции *у=sin(x)* для *0≤x≤180* c шагом в 10.
8. Подсчитать площади десяти кругов с радиусами от 1 см с шагом 2 см и вывести значения площадей на экран.
9. Вводятся положительные числа. Прекратить ввод чисел, когда их сумма превысит 100. Результат вывести на экран.
10. Вводятся числа. Прекратить ввод чисел, когда сумма положительных чисел превысит 100. Результат вывести на экран.
11. Вывести на экран значения произведений чисел a и b. Числа а изменяются от 1 до 11 с шагом 1, b – от 1 до 3 с шагом 0,2.
12. Вывести на экран таблицу перевода километров в мили в диапазоне от 2 до 20 километров с шагом 2 км.
13. Вы положили в банк 1500 рублей. Определить, сколько денег будет на Вашем вкладе через 1 год, если каждый месяц вклад увеличивается на 0.76 % от суммы предыдущего месяца.
14. Решив заняться легкой атлетикой, Вы пробежали в первый день 2 км. Сколько километров Вы пробежите за 2 недели, если каждый день Вы увеличиваете дистанцию на 10% от предыдущего дня?