

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

Государственное образовательное учреждение
высшего профессионального образования

“МАТИ” — РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. К. Э. ЦИОЛКОВСКОГО

Кафедра “Моделирование систем и информационные технологии”

**СЛУЧАЙНЫЕ ЧИСЛА, РЕКУРСИЯ, ОПЕРАТОР ВАРИАНТА,
ОПЕРАЦИИ СДВИГОВ И ФУНКЦИЯ ODD НА ПАСКАЛЕ**

Методические указания к лабораторной работе по курсу
“Алгоритмические языки и программирование”

Составитель В. В. Лидовский

Москва 2006

ВВЕДЕНИЕ

Настоящие методические указания предназначены для обеспечения учебного процесса студентов второго курса дневной формы обучения специальности 220200 “АСОИиУ” при выполнении лабораторной работы по предмету “Алгоритмические языки и программирование”.

Цель лабораторной работы: изучить простейшие методы программирования на языке Паскаль на примере реализации действий с псевдослучайными числами и прямой рекурсии, использования оператора варианта и операций поразрядных сдвигов, а также логической функции `odd`.

1. СИСТЕМНЫЕ ТРЕБОВАНИЯ

Для выполнения лабораторной работы необходимы следующие системные компоненты:

- а) компьютер, работающий под управлением операционной системы Linux;
- б) компилятор Free Pascal версии не ранее 2000 года.

2. МОДЕЛИРУЕМЫЕ ОБЪЕКТЫ И ДЕЙСТВИЯ

2.1. Случайные и псевдослучайные числа

Генерация настоящих случайных чисел — это весьма сложная, требующая специальной аппаратуры, дорогостоящая операция. Вместо нее поэтому, как правило, используют функции, генерирующие ряд чисел, зависимость между которыми очень сложно установить. Такие числа и называют *псевдослучайными*. Обычно каждое следующее число в ряду псевдослучайных чисел зависит от предыдущего. Главный недостаток таких чисел заключается в их периодичности, т. е. такие числа состоят из повторяющихся конечных последовательностей. Для хороших генерирующих функций этот период может быть весьма велик.

Средства для генерации последовательности псевдослучайных чисел в языках программирования высокого уровня, в частности Паскаля, состоят из двух стандартных подпрограмм: процедуры для инициализации генератора псевдослучайных чисел — она устанавливает первое число в последовательности; функции, которая возвращает очередное псевдослучайное число. В Паскале процедура называется `randomize`, а функция — `random`.

Средства типа процедуры `randomize` — необходимы, т. к. без них при каждом запуске программы генерировались бы одинаковые последовательности псевдослучайных чисел.

Функция `random` используется в Паскале в двух формах: без аргумента — для генерации вещественного псевдослучайного числа, строго

меньшего 1 и больше или равного 0; с положительным целым аргументом n — для генерации неотрицательного целого псевдослучайного числа, строго меньшего n . Таким образом, результатом `random(4)` может быть 0, 1, 2 или 3.

2.2. Рекурсия и итерация

Языки программирования имеют два эквивалентных функционально, но совершенно разных по форме средства для организации повторяющихся вычислений: операторы циклов, реализующие итерацию, и рекурсию. Итерационный способ организации повторов не требует, в отличие от рекурсии, дополнительной памяти и является незначительно более быстрым. Рекурсия описывается более компактно и не требует введения вспомогательных программных объектов, таких как переменные для циклов, что делает программные конструкции с ее использованием более наглядными. Рекурсия более соответствует декларативному в большинстве случаев характеру естественных языков, чем итерация.

Рассмотрим в качестве примера расчет факториала от числа n . Итерационный алгоритм описывается следующей последовательностью шагов:

- 1) $f \leftarrow 1$;
- 2) $i \leftarrow 1$;
- 3) пока $i < n$ вычисли $f \leftarrow f * i$ и $i \leftarrow i + 1$;
- 4) искомая величина равна f .

Рекурсивный алгоритм — это одно предложение: при $n < 2$ факториал равен 1, а иначе факториал равен n , умноженному на факториал от $n - 1$.

Рекурсия — это наиболее простой способ для вычисления *чисел Фибоначчи* (первые два таких числа равны 1, а остальные — это сумма двух предыдущих), которые часто встречаются в природе, например, в “родословном дереве пчел”. Количество спиралей в “корзинке” подсолнуха — это тоже числа Фибоначчи. При анализе чисел Фибоначчи естественно возникает *отношение золотого сечения* $\phi = (1 + \sqrt{5})/2 \approx 1.618$.

Эти числа имеют множество свойств. Например, если F_n — это n -е число Фибоначчи, то верно, что

$$F_{n+1}F_{n-1} - F_n^2 = (-1)^n \quad \text{и} \quad \text{НОД}(F_m, F_n) = F_{\text{НОД}(m,n)}.$$

2.3. Оператор варианта

Это синтаксически наиболее сложный оператор Паскаля — он позволяет осуществлять выбор одного из k операторов и функционально является обобщением условного оператора, для которого $k = 2$. Таким образом, если условный оператор использует булеву, бинарную логику, то оператор варианта использует обобщенную, k -значную.

2.4. Операции поразрядных сдвигов и функция проверки на нечетность

Операции поразрядных сдвигов являются одними из самых фундаментальных для цифровой вычислительной техники — они реализуются на аппаратном уровне любым компьютерным процессором.

Эти операции реализуют частные случаи деления и умножения. Сдвиг вправо (`shr`) соответствует делению на 2, а сдвиг влево (`shl`) — умножению на 2. Используя сдвиги и операции сложения и умножения, можно реализовать весьма быстрые алгоритмы для общего случая деления и умножения — так, например, эти операции реализовывались до середины 1990-х.

Операция-предикат `odd` определена на целых числах, ее результат истина для нечетных чисел и ложь для четных.

3. ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

3.1. Требования к структуре программы

Программа должна содержать рекурсивную функцию для расчета факториала, функцию с именем `Fib` для расчета чисел Фибоначчи и определяемую вариантом задания функцию f , а также и четыре функции с параметром-массивом для вычисления среднего квадратического, арифметического, геометрического и гармонического. При вычислении f нужно использовать операцию поразрядного сдвига вместо умножения на 2.

Программа должна решать следующие задачи:

- 1) подсчет средних (квадратического, арифметического, геометрического и гармонического) 100 случайных чисел из диапазона от 1 до 100;
- 2) вычисление $7!$, $10!$ и $12!$;
- 3) вычисление чисел Фибоначчи с номерами 6, 11 и 41;
- 4) вычисление значения функции f , заданной вариантом задания, в точках 4, 12, 16 и 32;
- 5) проверка на четность чисел Фибоначчи с номерами 13 и 15 (использовать функцию `odd`).

3.2. Анализ результатов

Проверить соответствие полученных результатов их теоретическим оценкам.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Почему простой рекурсивный алгоритм для вычисления чисел Фибоначчи так неэффективен?
2. Каковы отношения между полученными средними значениями?

3. Можно ли утверждать, что нерекурсивный алгоритм для вычисления факториала будет намного быстрее рекурсивного?
4. Можно ли заменять оператор варианта на несколько условных операторов?
5. Как должно выглядеть определение функции, эквивалентной odd?
6. Сколько раз нужно использовать процедуру randomize в программе?
7. Как может выглядеть определение функции для быстрого расчета чисел Фибоначчи?
8. Каково максимальное число Фибоначчи и максимальный факториал, которые можно вычислить на компьютере при использовании типа longint?
9. Можно ли использовать операцию сдвига вместо деления на 2?

5. ВАРИАНТЫ РАБОТ

Лабораторная работа по теме “Случайные числа, рекурсия, оператор варианта, операции сдвигов и функция odd на Паскале” имеет 2 варианта заданий, определяемых следующей таблицей.

| | |
|----|--|
| 1. | $f(n) = \begin{cases} 0, & \text{при } n \geq 30 \\ 2^n/\text{Fib}(n), & \text{при } 15 \leq n < 30 \\ n!/4^n, & \text{при } 10 \leq n < 15 \\ 8^n/3, & \text{при } n < 10 \end{cases}$ |
| 2. | $f(n) = \begin{cases} 1, & \text{при } n \geq 31 \\ 2^n/\text{Fib}(n), & \text{при } 15 \leq n < 31 \\ n!/4^n, & \text{при } 8 \leq n < 14 \\ n!/\text{Fib}(n), & \text{при } n < 7 \end{cases}$ |

Вещественная функция $f(n)$ определена на множестве неотрицательных целых чисел.

ЛИТЕРАТУРА

1. Бородин Ю.С., Вальвачев А.Н., Кузьмич А.И. *Паскаль для персональных компьютеров* — Минск: Вышэйшая школа, БФ ГИТМП “НИКА”, 1991. — 369 с.
2. Зуев Е.А. *Язык программирования Turbo Pascal 6.0* — М.: Унитех, 1992. — 298 с.
3. Грэхэм Р., Кнут Д.Н., Паташник О. *Конкретная математика* — М.: Мир, 1998. — 703 с.
4. Петерсен Р. *Linux: полное руководство* — Киев: “Ирина” ВНУ, 2000. — 642 с.

ОГЛАВЛЕНИЕ

| | стр. |
|--|------|
| Введение | 3 |
| 1. Системные требования | 3 |
| 2. Моделируемые объекты и действия | 3 |
| 2.1 Случайные и псевдослучайные числа | 3 |
| 2.2 Рекурсия и итерация | 4 |
| 2.3 Оператор варианта | 4 |
| 2.4 Операции поразрядных сдвигов и функция проверки на нечетность | 5 |
| 3. Порядок выполнения лабораторной работы | 5 |
| 3.1. Требования к структуре программы | 5 |
| 3.2. Анализ результатов | 5 |
| 4. Контрольные вопросы | 5 |
| 5. Варианты работ | 6 |
| Литература | 6 |

Владимир Викторович Лидовский

СЛУЧАЙНЫЕ ЧИСЛА, РЕКУРСИЯ, ОПЕРАТОР ВАРИАНТА,
ОПЕРАЦИИ СДВИГОВ И ФУНКЦИЯ ODD НА ПАСКАЛЕ

Методические указания к лабораторной работе по курсу
“Алгоритмические языки и программирование”

Редактор М. А. Соколова

Оригинал-макет подготовлен в пакете Plain-TeX