Мишанин

**Лабораторная работа №1.**

**Оценка временной сложности алгоритмов**

Целью лабораторной работы является приобретение навыков исследования временной сложности алгоритмов и определения ее асимптотических оценок.

* + 1. Требования к содержанию, оформлению и порядку выполнения

В содержательной части отчета по выполнению лабораторной работы требуется привести описание алгоритма, выбранного согласно своему варианту, провести его анализ и определить асимптотические оценки его временной сложности. Алгоритм рекомендуется оформлять с помощью блок-схем.

* + 1. Теоретическая часть

Теоретические сведения, необходимые для выполнения лабораторной работы, представлены в лекции.

* + 1. Общая постановка задачи

Требуется провести анализ и оценку временной сложности заданного алгоритма. Варианты заданий представлены в таблице в следующем разделе.

В качестве дополнительных заданий рекомендуется программно реализовать заданный алгоритм.

Варианты заданий к лабораторной работе № 1

|  |  |
| --- | --- |
| *Вариант* | *Алгоритм* |
| 1 | Тривиальный алгоритм возведения в степень (рис. 3.1) |
| 2 | Рекурсивный алгоритм возведения в степень (рис. 3.2) |
| 3 | Алгоритм быстрого возведения в степень (рис. 3.3 а) |
| 4 | Алгоритм быстрого возведения в степень (рис. 3.3 б) |
| 5 | Алгоритм вычисления значения многочлена (рис. 3.4) |
| 6 | Алгоритм вычисления значения многочлена по схеме Горнера (рис. 3.5) |
| 7 | Алгоритм сортировки обменом (рис. 4.5) |
| 8 | Алгоритм сортировки выбором (рис. 4.7) |
| 9 | Алгоритм сортировки вставками (рис. 4.9) |
| 10 | Алгоритм быстрой сортировки (рис. 4.14, 4.16) |

* + 1. Ход работы

Рассмотрим 6 вариант. Требуется провести анализ и оценку временной сложности алгоритм вычисления значения многочлена по схеме Горнера.

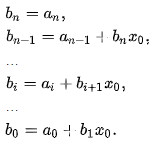
Алгоритм предназначен для решения следующей задачи: Алгоритм вычисления значения многочлена по схеме Горнера.

Задан многочлен

{\displaystyle P(x)=a\_{0}+a\_{1}x+a\_{2}x^{2}+a\_{3}x^{3}+\ldots +a\_{n}x^{n},\quad a\_{i}\in \mathbb {R} .} 

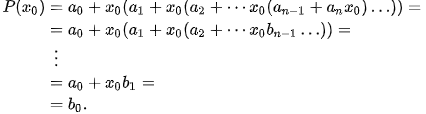
Пусть требуется вычислить значение данного многочлена при фиксированном значении {\displaystyle x=x\_{0}} . Представим многочлен {\displaystyle P(x)}  в следующем виде:

Определим следующую последовательность:

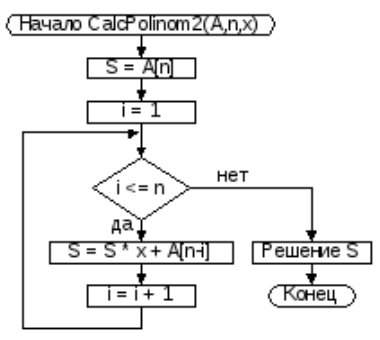


Искомое значение {\displaystyle P(x\_{0})}  есть {\displaystyle b\_{0}} . Покажем, что это так.

В полученную форму записи {\displaystyle P(x)}  подставим {\displaystyle x=x\_{0}}  и будем вычислять значение выражения, начиная с внутренних скобок. Для этого будем заменять подвыражения через  {\displaystyle b\_{i}}:

{\displaystyle {\begin{aligned}P(x\_{0})&=a\_{0}+x\_{0}(a\_{1}+x\_{0}(a\_{2}+\cdots x\_{0}(a\_{n-1}+a\_{n}x\_{0})\dots ))=\\&=a\_{0}+x\_{0}(a\_{1}+x\_{0}(a\_{2}+\cdots x\_{0}b\_{n-1}\dots ))=\\&~~\vdots \\&=a\_{0}+x\_{0}b\_{1}=\\&=b\_{0}.\end{aligned}}} 

Алгоритм вычисления значения многочлена по схеме Горнера основан на выше приведенных соображениях. Его блок-схема приведена на рис.



Подсчет временной сложности в этом случае гораздо проще: для вычисления Siтребуется 1 умножение и 1 сложение. Всего такая итерация осуществляетсяnраз. Таким образом, временная сложность этого алгоритма:Т(п) =nумножений +nсложений =2nопераций.

Зачастую такая подробная оценка временной сложности не требуется. Вместо нее приводят лишь асимптотическую скорость возрастания количества операций при увеличении n.

Так, например, функция Т(п) = n2/2 + 3n/2 + 1возрастает приблизительно какn2/2(отбрасываем сравнительно медленно растущее слагаемое3n/2+1). Константный множитель1/2также убираем и получаем асимптотическую оценку для алгоритма 1, которая обозначается специальным символомO(n2)(читается как "О большое от эн квадрат"). Это – верхняя оценка, т.е количество операций (а значит, и время работы) растет не быстрее, чем квадрат количества элементов. В этом случае говорят, чтоТ(п) есть O(n2), илиТ(п) имеет порядок O(n2).

* + 1. Контрольные вопросы к защите

1. Понятие временной сложности алгоритма.
2. Определение асимптотических оценок временной сложности.
3. Основные принципы получения асимптотических оценок.
4. *Правила анализа алгоритмов с целью определения их временной сложности.*

Контрольные вопросы

1. Временная сложность алгоритма это функция размера входных данных, равная максимальному количеству элементарных операций, проделываемых алгоритмов для решения экземпляра задачи указанного размера. Во многих задачах размер выхода не превосходит или пропорционален размеру входа, в этом случае можно рассматривать, временную сложность как функцию размера только входных данных.