

Алгоритмизация и программирование. Язык C++

- § 38. Целочисленные алгоритмы
- § 39. Структуры
- § 40. Динамические массивы
- § 41. Списки
- § 42. Стек, очередь, дек
- § 43. Деревья
- § 44. Графы
- § 45. Динамическое программирование

Алгоритмизация и программирование. Язык C++

§ 38. Целочисленные алгоритмы

Решето Эратосфена



Эратосфен Киренский
(Eratosthenes, Ερατοσθένης)
(ок. 275-194 до н.э.)

Алгоритм:

- 1) начать с $k = 2$
- 2) «выколоть» все числа через k , начиная с $k \cdot k$
- 3) перейти к следующему «невыколотому» k
- 4) если $k \cdot k \leq N$, то перейти к шагу 2
- 5) напечатать все числа, оставшиеся «невыколотыми»

Новая версия – [решето Аткина](#).

? Как улучшить?

+ высокая скорость, количество операций

$$O((N \cdot \log N) \cdot \log \log N)$$

- нужно хранить в памяти все числа от 1 до N

Решето Эратосфена

Задача. Вывести все простые числа от 2 до N .

Объявление переменных:

```
const int N=100;  
bool A[N+1];  
int i, k;
```

выделяем на 1
элемент больше,
чтобы начать с $A[1]$

Сначала все невычеркнуты:

```
for ( i = 2; i <= N; i++ )  
    A[i] = true;
```

Решето Эратосфена

Вычёркивание непростых:

```
k = 2;
while ( k*k <= N ) {
    if ( A[k] ) {
        i = k*k;
        while ( i <= N )
        {
            A[i] = false;
            i += k;
        }
    }
    k ++;
}
```

Решето Эратосфена

Вывод результата:

```
for ( i = 2; i <= N; i++ )  
    if ( A[i] )  
        cout << i << " " ;
```

«Длинные» числа

Ключи для шифрования: ≥ 256 битов.

Целочисленные типы данных: ≤ 64 битов.



Как хранить?

Длинное число – это число, которое не помещается в переменную одного из стандартных типов данных языка программирования.

«**Длинная арифметика**» – алгоритмы для работы с длинными числами.

«Длинные» числа

A = 12345678

	0	1	2	3	4	5	6	7	8	9
A	1	2	3	4	5	6	7	8	0	0

? Что плохо?



- нужно хранить длину числа
- неудобно вычислять (с младшего разряда!)
- неэкономное расходование памяти

Обратный порядок элементов:

	9	8	7	6	5	4	3	2	1	0
A	0	0	1	2	3	4	5	6	7	8

«Длинные» числа

Упаковка элементов: $A = 12345678$

	9	8	7	6	5	4	3	2	1	0
A	0	0	0	0	0	0	0	12	345	678

$$12345678 = 12 \cdot 1000^2 + 345 \cdot 1000^1 + 678 \cdot 1000^0$$



На что похоже?

система счисления с
основанием 1000!

`long int`:

от $-2^{31} = -2\,147\,483\,648$ до $2^{31} - 1 = 2\,147\,483\,647$.



Какие основания можно использовать?

должны помещаться все
промежуточные результаты!

Вычисление факториала

Задача 1. Вычислить точно значение факториала

$$100! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot 99 \cdot 100$$

? Как оценить количество цифр?

$$1 \cdot 2 \cdot 3 \cdot \dots \cdot 99 \cdot 100 <$$

201 цифра

основание 1000000

6 цифр в ячейке \Rightarrow 34 ячейки

```
const int N = 33;  
long int A[N+1];
```

Основной алгоритм:

длинное
число

```
[A] = 1;  
for ( k = 2; k <= 100; k ++ )  
    [A] = [A] * k;
```

Вычисление факториала

основание $d = 1\ 000\ 000$

$[A] = 12345678901734567$

	3	2	1	0	
A	0	12345	678901	734567	*3

$734\ 567 \cdot 3 = 2\ 203\ 701$

остаётся в A[0]

r = перенос в A[1]

? Как найти перенос?

```
s = A[0] * k;
A[0] = s % d;
r = s / d;
```

? Что изменится для A[1]?

$s = A[1] * k + r;$

Вычисление факториала

Умножение «длинного» числа на k :

```
r = 0;
for ( i = 0; i <= N; i++ ) {
    s = A[i] * k + r;
    A[i] = s % d;
    r = s / d;
}
```

все разряды

Вычисление $100!$:

```
for ( k = 2; k <= 100; k++ )
{
    ...
}
```

Вывод длинного числа

	3	2	1	0
A	0	1	2	3



Какое число?

[A] = 1000002000003

- найти старший ненулевой разряд

```
i = N;  
while ( ! A[i] )  
    i --;
```

- вывести этот разряд

```
cout << A[i];
```

- вывести все следующие разряды, добавляя лидирующие нули до 6 цифр

Вывод длинного числа

Вывод остальных разрядов:

```
for ( k=i-1; k >= 0; k-- )
    Write6 ( A[k] );
```

со старшего

Write6:

$x = 12345$

$x / 100000$

012345

$x \% 100000$

x	M	x / M
12345	100000	0

Вывод длинного числа

Вывод числа с лидирующими нулями:

```
void Write6 ( long int x )
{
    long int M=100000;
    while ( M > 0 )
    {
        cout << x / M;
        x %= M;
        M /= 10;
    }
}
```

Конец фильма

ПОЛЯКОВ Константин Юрьевич

д.т.н., учитель информатики

ГБОУ СОШ № 163, г. Санкт-Петербург

kpolyakov@mail.ru

ЕРЕМИН Евгений Александрович

к.ф.-м.н., доцент кафедры мультимедийной

дидактики и ИТО ПГГПУ, г. Пермь

eremin@pspu.ac.ru

Источники иллюстраций

1. wallpaperscraft.com
2. www.mujerhoy.com
3. www.pinterest.com
4. www.wayfair.com
5. www.zchocolat.com
6. www.russiantable.com
7. www.kursachworks.ru
8. ebay.com
9. centrgk.ru
10. www.riverstonellc.com
11. 53news.ru
12. 10hobby.ru
13. ru.wikipedia.org
14. иллюстрации художников издательства «Бином»
15. авторские материалы