



Арифметические задачи

- поиск решения – «тупой» (полный) перебор
- **эвристический алгоритм** - определение свойств, зависимостей, которые сокращают поиск (без компьютера – вплоть до единственного)
- разложение на цифры
- проверка условий

```
#include <stdio.h>
// 29. Задача о десяти цифрах. Расставьте все десять цифр 1, 2, 3, 4, 5, 6, 7, 8, 9, 0
// в таком порядке, чтобы получившееся число делилось на все числа от 2 до 18.
int main() {
    printf("a-a-a-a-a-a");
}
```

- цикл перебора, оценка времени

```
8
delay = 7687
```

```
#include <stdio.h>
#include <time.h>
// 29. Задача о десяти цифрах. Расставьте все десять цифр 1, 2, 3, 4, 5, 6, 7, 8, 9, 0
// в таком порядке, чтобы получившееся число делилось на все числа от 2 до 18.
int main() {
    long long a = 0;
    printf("%d\n", sizeof(a)); // Проверка размерности 10^10 -> 10 * 2^30 > 2^34
    long long t0 = clock(); // Время "от сотворения мира" (1.1.1970 ???) в мс
    for (a = 0; a < 10000000000; a++) { // Полный перебор
        }
    int delay = clock() - t0; // long - long -> int
    printf("delay = %d\n", delay);
}
```



Арифметические задачи

- проверка на делимость 2...18 – паттерн ДЛЯ ВСЕХ
- предупреждение – размерность константы (С «вытягивает»)
- ошибка – формат спецификатора в printf

```
#include <stdio.h>
#include <time.h>
// 29. Задача о десяти цифрах. Расставьте все десять цифр 1, 2, 3, 4, 5, 6, 7, 8, 9, 0
// в таком порядке, чтобы получившееся число делилось на все числа от 2 до 18.
int main() {
    long long a = 0;
    printf("%d\n", sizeof(a));           // Проверка размерности 10^10 -> 10 * 2^30 > 2^34
    long long t0 = clock();              // Время "от сотворения мира" (1.1.1970 ???) в мс
    for (a = 0; a < 100000000000LL; a++) { // Полный перебор
        // проверка делимости - паттерн ДЛЯ ВСЕХ
        int bad = 0;
        for (int n = 2; n <= 18; n++)
            if (a % n != 0) {             // Не делится
                bad = 1;
                break;
            }
        if (!bad) {                       // Проверка ПОСЛЕ
            printf("value = %lld\n", a);
        }
    }
    int delay = clock() - t0;             // long - long -> int
    printf("delay = %d\n", delay);
}
```

```
value = 9948818880
value = 9961071120
value = 9973323360
value = 9985575600
value = 9997827840
delay = 197580
```

- время – 197 с
- эвристики – на 10 – последняя цифра = 0

```
value = 9961071120
value = 9973323360
value = 9985575600
value = 9997827840
delay = 31192
```

```
for (a = 0; a < 100000000000LL; a+=10) { // Полный перебор
    // проверка делимости - паттерн ДЛЯ ВСЕХ
```



Арифметические задачи

- сумма цифр: время – 210 с
- делимость на 9, подсчет результатов

```
int num = 0;
for (a = 0; a < 10000000000L; a+=10) { // Полный перебор
    // Сумма цифр
    int sum = 0;
    for (long long b = a; b != 0; b = b / 10)
        sum += b % 10;
    if (sum % 9 != 0)
        continue;
    // проверка делимости – паттерн для ВСЕХ
    int bad = 0;
    for (int n = 2; n <= 18; n++)
        if (a % n != 0) { // Не делится
            bad = 1;
            break;
        }
    if (!bad) { // Проверка ПОСЛЕ
        num++;
        printf("num=%d value = %lld\n", num, a);
    }
}
```

```
int sum = 0;
for (long long b = a; b != 0; b = b / 10)
    sum += b % 10;
if (sum % 9 != 0)
    continue;
// Число делится на 4
if (a / 10 % 10 % 2 != 0) {
    continue;
}
// проверка делимости – паттерн для ВСЕХ
int bad = 0;
// уже делится на 10, 2, 5, 3, 6, 9 и 4
for (int n = 7; n <= 18; n++)
```

```
value = 9948818880
value = 9961071120
value = 9973323360
value = 9985575600
value = 9997827840
delay = 210170
```

```
num=811 value = 9924314400
num=812 value = 9936566640
num=813 value = 9948818880
num=814 value = 9961071120
num=815 value = 9973323360
num=816 value = 9985575600
num=817 value = 9997827840
delay = 185624
```

число делится на 4, если:

- в его последнем разряде стоят цифры 0, 4 или 8, а предпоследний разряд при этом является четным;

```
num=813 value = 9948818880
num=814 value = 9961071120
num=815 value = 9973323360
num=816 value = 9985575600
num=817 value = 9997827840
delay = 184486
```

Замечание: проверку делимости на 9 можно не делать, это в условиях задачи ($\sum 1..9 = 45$)



Арифметические задачи

- повторение цифр
- вариант 2 – массив индикаторов $\text{ind}[i]=1$ – была цифра i
- вариант 1 - разряды машинного слова (поразрядные операции)

```
// Сумма цифр и цифры разные
int sum = 0;
int mask = 0;
int bad2 = 0;
for (long long b = a; b != 0; b = b / 10) {
    int k = b % 10;           // Цифра
    int mm = 1 << k;          // k-ый разряд mask
    if ((mask & mm) != 0) {    // разряд уже занят
        bad2 = 1;
        break;
    }
    mask |= mm;               // Установить разряд
    sum += k;
}
if (bad2)                    // Есть повторения цифр
    continue;
if (sum % 9 != 0)
    continue;
```

Debug (x86)

```
num=1 value = 0
num=2 value = 2438195760
num=3 value = 3785942160
num=4 value = 4753869120
num=5 value = 4876391520
delay = 85502
```

Release (x64)

```
num=1 value = 0
num=2 value = 2438195760
num=3 value = 3785942160
num=4 value = 4753869120
num=5 value = 4876391520
delay = 10430
```

```
// Сумма цифр и цифры разные
int sum = 0;
int bad2 = 0;
int mask[10];
for (int i = 0; i < 10; i++)
    mask[i] = 0;
for (long long b = a; b != 0; b = b / 10) {
    int k = b % 10;           // Цифра
    if (mask[k] != 0) {       // Цифра уже была
        bad2 = 1;
        break;
    }
    mask[k] = 1;              // Установить признак
    sum += k;
}
```

Debug (x86)

```
num=1 value = 0
num=2 value = 2438195760
num=3 value = 3785942160
num=4 value = 4753869120
num=5 value = 4876391520
delay = 89466
```

Release (x64)

```
num=1 value = 0
num=2 value = 2438195760
num=3 value = 3785942160
num=4 value = 4753869120
num=5 value = 4876391520
delay = 10634
```

Имя	ИД п...	Состояние	Имя польз...	ЦП
Бездействие системы	0	Выполняется	СИСТЕМА	78
YandexDisk2.exe	8292	Выполняется	Admin	08
fufuVS22.exe	17248	Выполняется	Admin	08
msedgewebview2.exe	17000	Выполняется	Admin	01



Арифметические задачи

Рекурсивный перебор допустимых сочетаний цифр $9! = 362\,880$ вариантов (сprog)

```
// Последовательность без повторений CPROG 74-01
void F3(int W[], int R[], int n, int N, int& cnt) {
    int i;
    if (n == N) {
        cnt++;
        //printf("\n");
        //for (i = 0; i < n; i++) printf("%d ", W[i]);
        return;
    }
    for (i = 0; i < N; i++) {           // цикл по всем оставшимся
        if (R[i] == 0) continue;       // пропуск уже выбранного
        W[n] = R[i];                   // выбор оставшегося
        R[i] = 0;                      // исключение из исходного
        F3(W, R, n + 1, N, cnt);
        R[i] = W[n];                   // возвращение выбранного
    }
}

int main() {
    int in[9] = { 1,2,3,4,5,6,7,8,9 };
    int res[9];
    int cnt = 0;
    long long t0 = clock();            // Время "от сотворения мира"
    F3(res, in, 0, 9, cnt);
    printf("cnt=%d ", cnt);
    int delay = clock() - t0;           // long - long -> int
    printf("delay = %d\n", delay);
}
```

Release (x64)

```
cnt=362880 delay = 9
```



Арифметические задачи

Рекурсивный перебор допустимых сочетаний цифр $9! = 362\,880$ вариантов (сprog)

Release (x64)

```
// Последовательность без повторений CPROG 74-01
void F3(int W[], int R[], int n, int N, int& cnt) {
    int i;
    if (n == N) {
        // Составить число из последовательности цифр и проверить
        long long a = 0;
        for (int i = 0; i < 9; i++)
            a = a * 10 + W[i];
        a = a * 10;
        int bad = 0;
        for (int n = 7; n <= 18; n++)
            if (a % n != 0) {                // Не делится
                bad = 1;
                break;
            }
        if (!bad) {                          // Проверка ПОСЛЕ
            cnt++;
            printf("num=%d value = %lld\n", cnt, a);
        }
        return;
    }
    for (i = 0; i < N; i++) {                // цикл по всем оставшимся
        if (R[i] == 0) continue;            // пропуск уже выбранного
        W[n] = R[i];                        // выбор оставшегося
        R[i] = 0;                           // исключение из исходного
        F3(W, R, n + 1, N, cnt);            // возвращение выбранного
        R[i] = W[n];
    }
}
```

```
num=1 value = 2438195760
num=2 value = 3785942160
num=3 value = 4753869120
num=4 value = 4876391520
cnt=4 delay = 14
```