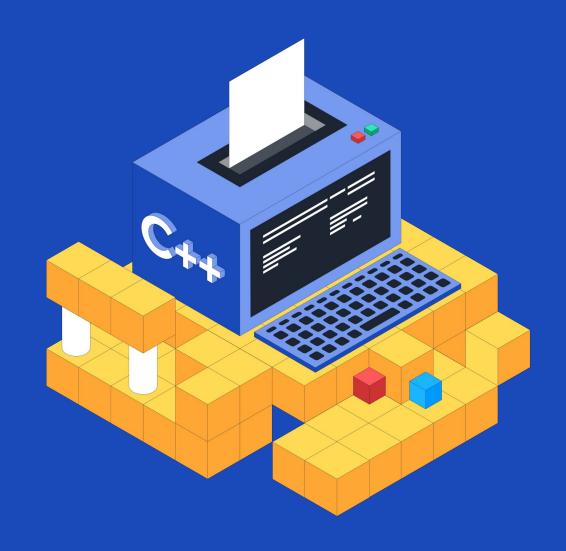




ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ C++

Материалы подготовлены отделом методической разработки

Средний уровень







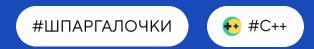


Перевод чисел из одной

системы счисления в



другую





Системы счисления

Система счисления - это способ составить и записать число из отдельных цифр.

Основание системы счисления - это количество цифр, используемых в ней для записи чисел.

В привычной нам десятичной системе счисления 10 цифр:

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

В двоичной - всего 2 цифры:

0, 1

А в шестнадцатеричной - целых 16. Цифры больше 9 заменяют буквы латинского алфавита:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

Существуют и другие системы счисления, например, восмеричная.







Системы счисления

двоичная: 01

восьмеричная: 01234567

десятичная: 0123456789

шестнадцатеричная: 0 1 2 3 4 5 6 7 8 9 A B C D E F







Алгоритм перевода числа из десятичной системы счисления

Десятичное число легко перевести в любую другую систему счисления. Алгоритм следующий: нужно делить это число на основание искомой системы счисления, пока частное не станет равно нулю. Остаток от каждого деления нужно сохранять. В конце остается записать остатки от всех делений, от последнего к первому. Получившееся число и будет результатом.

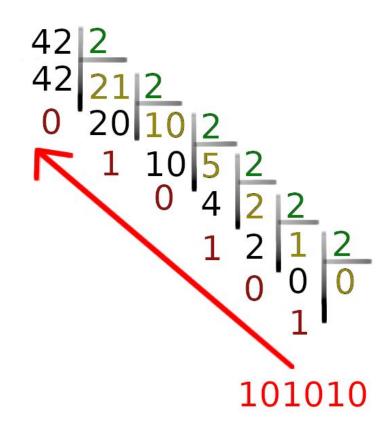






Алгоритм перевода числа из десятичной системы счисления в двоичную





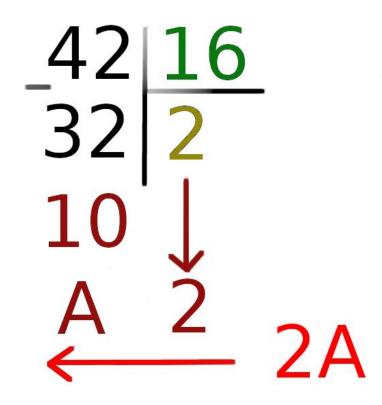




СЯТИЧНОЙ

Алгоритм перевода числа из десятичной системы счисления в шестнадцатеричную











```
string result = "";
char hexSimbols[6] = {'A', 'B', 'C', 'D', 'E', 'F'};
while (number > 0) {
     int remainder = number % radix;
     if (remainder > 9) {
          result = hexSymbols[remainder - 10] + result;
     else {
          result = to_string(remainder) + result;
     number /= radix;
```







Алгоритм перевода числа в десятичную систему счисления

Перевод числа из любой другой системы счисления в десятичную требует работы со степенями. Число нужно разложить на цифры и каждой из них присвоить степень, начиная с 0. Степень увеличивается справа налево (от последней цифры числа к первой).

Далее каждую цифру числа нужно умножить на основание системы счисления, возведенное в полученную степень, а результаты сложить. Полученное число и будет искомым десятичным числом.

Если при разложении числа на цифры встречается 0, вычисления с ним можно не производить, но при определении степени его нужно учитывать.





Алгоритм перевода числа из двоичной системы счисления в десятичную

$$101010$$

$$1^{5} \quad 0^{4} \quad 1^{3} \quad 0^{2} \quad 1^{1} \quad 0^{0}$$

$$1 \times 2^{5} + 0 \times 2^{4} + 1 \times 2^{3} + 0 \times 2^{2} + 1 \times 2^{1} + 0 \times 2^{0}$$

$$1 \times 32 + 1 \times 8 + 1 \times 2$$

$$32 + 8 + 2 = 42$$







Алгоритм перевода числа из шестнадцатеричной системы счисления в десятичную







Прежде, чем писать код для перевода числа из другой системы счисления в десятичную, нужно подготовить вспомогательные функции.

Исходное число хранится в строке и может содержать буквы (если оно шестнадцатеричное), а значит нужна функция, которая проверит, содержит символ цифру или букву. А также функции для перевода символа в целое число и для получения индекса элемента из массива.







Функция, проверяющая, является ли символ цифрой (находится ли он между символами '0' и '9'). bool isInteger(char n) { if (n >= '0' && n <= '9') { return true; return false;







```
Функция, преобразующая символ в число (из кода полученного символа вычитает код символа '0'): int charToInt(char n) { return int(n) - int('0'); }
```







```
Функция, возвращающая индекс элемента в массиве:
int getIndex(char n, char *array, int size) {
   for (int i = 0; i < size; i++) {
       if (array[i] == n) {
           return i; iii HELLOWORLD
    return -1;
```







```
int result = 0;
char hexSimbols[6] = {'A', 'B', 'C', 'D', 'E', 'F'};
int degree = number.length - 1; // последний индекс и первая слева степень совпадают
for (int i = 0; i < number.length(); i++) {
      char currentChar = number[i]; // текущий символ числа
      int currentNumber;
      if (isInteger(currentChar)) {
           currentNumber = charToInt(currentChar);
      else {
           currentNumber = getIndex(currentChar, hexSimbols, 6) + 10;
      result += currentNumber * pow(radix, degree);
      degree--;
```







Двоично-десятичный код

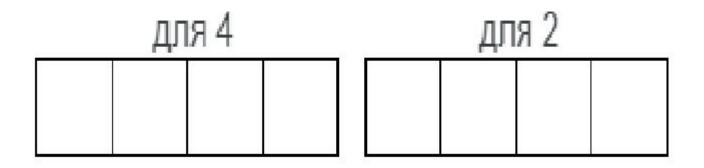
Двоично-десятичный код - один из способов хранения чисел, используется многими электронными приборами. Для преобразования числа в двоично-десятичный код нужно каждую его цифру преобразовать в двоичный код. Если в итоговом числе меньше 4 символов - дополнить нулями слева и записать полученные двоичные числа слева направо.







Пример для числа 42:



4 = 100 в двоичной с.с. 2 = 10 в двоичной с.с.

	1	0	0
--	---	---	---



0	1	0	0
---	---	---	---