

ТЕСТ ПО ИНФОРМАТИКЕ

1. Дана строка *abacabaabracadabra*, в которой используется 5 различных букв. Также заданы 5 кодов: 0, 11, 101, 1000, 1001. Требуется сопоставить буквы с кодами таким образом, чтобы при кодировании строки *abacabaabracadabra* итоговая длина кода была минимально возможной. Какой минимальной длины может оказаться итоговый код?

- (a) 28
- (b) 35
- (c) 36
- (d) 41
- (e) 44
- (f) 47
- (g) 49
- (h) 56

2. Возьмем все степени числа 2 от 1 до 2^{16} , все степени числа 4 от 1 до 4^{16} , а также все степени числа 8 от 1 до 8^{16} . Объединим все эти числа в один массив и отсортируем. Какое число находится на позиции 31? Позиции в массиве нумеруются с 1.

- (a) 512
- (b) 1024
- (c) 2048
- (d) 4096
- (e) 8192
- (f) 16384
- (g) 32768
- (h) 65536

3. Дан массив *a* длины *n* (элементы массива нумеруются с 0). Также задана функция *f(pos)* следующим образом:

Java

```
1 void f(int pos) {
2     System.out.println(pos);
3     pos = pos + a[pos];
4     if (pos < 0 || pos >= n) {
5         pos = n - 1;
6     }
7     f(pos);
8 }
```

C++

```
1 void f(int pos) {
2     std::cout << pos;
3     pos = pos + a[pos];
4     if (pos < 0 || pos >= n) {
5         pos = n - 1;
6     }
7     f(pos);
8 }
```

Python

```
1 def f(pos):
2     print(pos)
3     pos = pos + a[pos]
4     if (pos < 0) or (pos >= n):
5         pos = n - 1
6     f(pos)
```

Pascal

```
1 procedure f(pos: integer);
2 begin
3     writeln(pos);
4     pos := pos + a[pos];
5     if (pos < 0) or (pos >= n) then
6     begin
7         pos := n - 1;
8     end;
9     f(pos);
10 end;
```

При каком из следующих массивов выполнение $f(0)$ выведет на экран максимальное количество **уникальных** чисел?

- (a) [1, -1, 0, -4, 2, 2, -1]
- (b) [2, -1, 0, -1, 2, -4, 1]
- (c) [0, -1, 1, 2, 2, -1, -4]
- (d) [1, 2, -4, 2, 0, -1, -1]
- (e) [-4, -1, 2, 1, 2, 0, -1]
- (f) [0, -4, 1, -1, 2, -1, 2]
- (g) [-4, 2, 1, 2, 0, -1, -1]
- (h) [2, 2, -1, -1, 1, 0, -4]

4. Возьмем 4 натуральных числа a, b, c, d , у которых в двоичной записи ровно 10^{11} бит (учитывая лидирующие нули), биты пронумерованы целыми числами от 1 до 10^{11} . У числа a только каждый 3-й бит равен 1 (то есть биты под номером 3, 6, 9 и т.д.), у числа b — каждый 7-й, у числа c — каждый 4-й, у числа d — каждый 5-й. Требуется узнать, сколько битов 1 содержится в значении выражения $a \oplus b \oplus c \oplus d$, где \oplus является операцией побитового исключающего «ИЛИ».

- (a) 238095238
- (b) 3571428570
- (c) 19047619050
- (d) 42857142856
- (e) 46428571426
- (f) 53571428574
- (g) 65714285714
- (h) 99761904762

5. Рассмотрим массив из 5 чисел, каждое из которых представлено в двоичной системе счисления. Назовем массив *хорошим*, если его элементы можно переставить таким образом, что каждый элемент (кроме первого) отличается от предыдущего **ровно** в одном бите.

Например, массив [1000, 0100, 1001, 0101, 1101] является хорошим, так как после перестановки элементов можно получить массив [1000, 1001, 1101, 0101, 0100], где каждый элемент (кроме первого) отличается от предыдущего ровно в одном бите.

Какой из следующих массивов является хорошим?

- (a) [0000, 1010, 1001, 1101, 0111]
- (b) [0000, 1010, 1110, 0101, 1011]
- (c) [1001, 1010, 1011, 0011, 0111]
- (d) [0000, 0100, 1100, 0010, 1011]
- (e) [1010, 0110, 1001, 0101, 1101]
- (f) [0010, 1010, 0101, 0011, 0111]
- (g) [1000, 1100, 0010, 0110, 0101]
- (h) [1000, 0100, 1110, 0001, 0111]

6. Рассмотрим массив a нечетной длины n . Элементы массива пронумерованы целыми числами от 1 до n , и i -й элемент массива имеет значение i .

Также задана функция $f(pos)$, которая разворачивает подотрезок массива a с позиции pos до позиции $n - pos + 1$. Например, при $n = 9$, после выполнения $f(3)$ массив a будет выглядеть следующим образом: [1, 2, 7, 6, 5, 4, 3, 8, 9].

Последовательно выполнили $f(1), f(2), f(3), \dots, f(\frac{n-1}{2})$. Требуется найти количество различных пар таких позиций (i, j) , что $i < j$ и $a_i > a_j$, при $n = 10^5 + 1$.

- (a) 0
- (b) 50000
- (c) 100000
- (d) 100001
- (e) 2500075000
- (f) 5000050000
- (g) 10000000000
- (h) 10000100000