

Задание 1

Бельчонок хочет узнать, какой у Совёнка автобусный (из шести цифр) билет – счастливый или нет. Совёнок не хочет говорить ему напрямую. Если Бельчонок назовёт местоположение двух цифр на билете, Совёнок ему скажет разность между ними. Например, если это был билет 624390, то на запрос Бельчонка о 2 и 3 цифре Совёнок ответит -2. Опишите, каким способом за наименьшее число шагов можно узнать, является ли билет счастливым, и докажите, что за меньшее число шагов этого сделать нельзя.

Ответ: Спросить про 1 и 4, 2 и 5, 3 и 6. Счастливый билетик счастливый, когда $a1+a2+a3=a4+a5+a6 \Rightarrow (a1-a4)+(a2-a5)+(a3-a6) = 0$. Следовательно, сложив три полученных числа, мы узнаем ответ – если 0, то счастливый, если нет, то нет.

За меньшее число шагов невозможно, т.к. не узнаем информацию о двух цифрах.

Задание 2

Вычислите, сколько единичек в двоичной записи десятичного числа $(16^{16} + 256^6 - 1) \cdot (2^8) + 63$.

Ответ: Число $16^{16} = 2^{64}$ в двоичной записи выглядит вот так $10\dots 0$ с 64 нулями. В числе $256^6 = 2^{48}$ так же, но с 48 нулями. В их сумме в двоичной записи будет уже 2 единицы. Отняв 1, мы превратим $10\dots 0$ с 48 нулями в $1\dots 1$ с 48 единицами. Умножая на 2^8 , добавляем 8 нулей в конец. Прибавляя $63 = 1 + 2 + 4 + 8 + 16 + 32$, на место 6 нулей встанут 6 единиц. Итого 55 единиц.

Задание 3

Есть 3 восьмизначных числа в двоичной системе счисления (число 12, например, пишем так: 00001100). Бельчонок и Совёнок делают побитовое исключающее ИЛИ над первым и вторым числом, затем над вторым и третьим и, наконец, над первым и третьим. У Бельчонка получились соответственно 11000111 (1, 2), 10101010 (2, 3) и 01100111 (1, 3), у Совёнка 11000111 10101010 и 01101101. Известно, что кто-то из них написал полностью правильно, а другой ошибся. Узнайте, кто ошибся и напишите первое из чисел, если известно, что третье число делится на 8, но не делится на 16, а второе больше 240?

Ответ: При побитовом исключающем ИЛИ от трёх цифр мы можем получить 4 комбинации: 000 от 000 или 111, 011 от 001 или 110, 110 от 010 или 101, 101 от 011 или 100. Следовательно, Бельчонок совершил ошибку, ибо у него предпоследние цифры в числах все единички, что невозможно. Мы можем по шифровке Совёнка найти зашифрованное, но не точно. Нужно ещё и выбрать одну из двух возможных расшифровок. Дополнительная информация о втором и третьем числе помогает узнать последние 4 цифры третьего (1000) и первые 4 цифры второго (1111), и по ним однозначно дешифровать первое число, равное 00110101.

Задание 4

Какой пароль будет тяжелее взломать (то есть, в каком случае больше вариантов потребуется перебрать): восьмизначный пароль из символов русского алфавита (включая Ё), в которых ни одна из букв не может повторяться дважды, или же 15-значный пароль из 6 символов, в котором они могут повторяться?

Ответ: Первый. Количество комбинаций в нём $33 \cdot 32 \cdot 31 \cdot 30 \cdot 29 \cdot 28 \cdot 27 \cdot 26 > 3^8 \cdot (11 \cdot 2^5 \cdot 31 \cdot 10 \cdot 29 \cdot 26) = 3^8 \cdot 2^7 \cdot (11 \cdot 31 \cdot 5 \cdot 29 \cdot 13)$ (мы заменили 28 на 27, уменьшив число и тем самым вынесли больше за скобки). Во втором количество комбинаций $(2 \cdot 3)^{15}$. Сокращаем и остаётся сравнить $11 \cdot 31 \cdot 5 \cdot 29 \cdot 13$ и $2^8 \cdot 3^7$. Подсчётом можно увидеть, что они равны 642785 и 559872 соответственно.

Задание 5

Однажды Бельчонок, прогуливаясь, увидел целую аллею столбов, стоящих вдоль прямой дороги. Так как в лесу скоро должен наступить праздник, то эти столбы он захотел украсить праздничной мишурой. Но у Бельчонка очень мало мишуры, и ему нужно украсить все столбы. Напишите программу на любом языке программирования, которая позволит рассчитать наименьшую длину мишуры, требуемой для украшения столбов. Мишура подвешивается только между соседними столбами. Каждый столб должен быть соединён мишурой хотя бы с одним. Если мишуры не хватит, программа должна выдать ответ NO.

Входные данные:

Сначала вводится N – количество столбов и M – длина имеющейся у Бельчонка мишуры. Затем вводятся координаты столбов (различные неотрицательные целые числа) в количестве N .

Выходные данные: минимальная суммарная длина мишуры, либо NO, если минимальная длина больше имеющихся у Бельчонка запасов.

Пример

№	Ввод	Вывод
1	4 10 17 2 4 15	4
2	7 20 43 15 2 20 4 41 3	9
3	4 20 99 3 1 5	NO

Ответ: Сначала сортируем массив координат, затем пользуемся методом динамического программирования. От двух точек ответом будет разность их координат ($S_2 = A_2 - A_1$), от более чем двух – разность координат между последними столбами (ибо последний столб так или иначе придётся соединять с предпоследним) плюс $\min(S_{k-1}, S_{k-2})$.

Код в псевдоалгоритмическом языке:

Заводим числа N и M , массивы a и S

Считываем N и M

От $i = 1$ до N считываем в массив a

Сорт($a, a+n$); //возможна любая сортировка

От $i = 2$ до N : если $i = 2$, то $S[2] = a[2] - a[1]$

иначе $S[i] = a[i] - a[i-1] + \min(S[i-1], S[i-2])$

Если $S[N] \geq M$ то вывести $S[N]$

иначе NO

Вариант 2

Задание 1

Бельчонок хочет узнать, какой у Совёнка автобусный билет (из шести цифр) – с чётной суммой цифр или нет. Совёнок не хочет говорить ему напрямую. Если Бельчонок назовёт местоположение двух цифр на билете, Совёнок ему скажет разность между ними. Например, если это был билет 624390, то на запрос Бельчонка о 2 и 3 цифре Совёнок ответит -2. Опишите, каким способом за наименьшее число шагов можно узнать, является ли сумма цифр билета чётной, и докажите, что за меньшее число шагов этого сделать нельзя.

Ответ: Спросить про 1 и 4, 2 и 5, 3 и 6. Сумма цифр чётна, когда $a_1 + a_2 + a_3 = a_4 + a_5 + a_6$ по модулю 2 $\Rightarrow (a_1 - a_4) + (a_2 - a_5) + (a_3 - a_6) = 0$ по модулю 2. Следовательно, сложив три полученных числа, мы узнаем ответ – если сумма чётна, и сумма цифр в билете чётна.

За меньшее число шагов невозможно, т.к. не узнаем информацию о двух цифрах.

Задание 2

Вычислите, сколько единичек в троичной записи десятичного числа $3^{40} + 3^{20} - 365$.

Ответ: Ответ равен 7,1 в самом начале числа и 6 в конце.

$365 = 1 + 1 + 3 + 9 + 27 + 81 + 243$, то есть после его удаления мы получаем число 1000...02222..2111111

Задание 3

Есть 3 восьмизначных числа в двоичной системе счисления (число 12, например, пишем так: 00001100). Бельчонок и Совёнок делают побитовое исключающее ИЛИ над первым и вторым числом, затем над вторым и третьим и, наконец, над первым и третьим. У Бельчонка получились соответственно ответы 11011001 (1, 2), 10111010 (2, 3) и 01100011 (1, 3), у Совёнка 11011001, 10111011 и 01100001. Известно, что кто-то из них написал все ответы полностью правильно, а другой ошибся. Узнайте, кто ошибся и напишите первое из чисел, если известно, что третье число можно представить в виде $16n+9$, а второе больше 16, но меньше 32.

Ответ: При побитовом исключающем ИЛИ от трёх цифр мы можем получить 4 комбинации: 000 от 000 или 111, 011 от 001 или 110, 110 от 010 или 101, 101 от 011 или 100. Следовательно, Совёнок совершил ошибку, ибо у него предпоследние цифры в числах все единички, что невозможно.

Мы можем по шифровке Бельчонка найти зашифрованное, но не точно. Нужно ещё и выбрать одну из двух возможных расшифровок. Дополнительная информация о втором и третьем числе помогает

узнать последние 4 цифры третьего (1001) и первые 4 цифры второго (001), и по ним однозначно дешифровать первое число, равное 11001010.

Задание 4

Какой пароль будет тяжелее взломать (то есть, в каком случае больше вариантов потребуется перебрать): девятизначный пароль из символов русского алфавита (включая Ё), в которых ни одна из букв не может повторяться дважды, или же 17-значный пароль из 6 символов, в котором они могут повторяться?

*Ответ: Второй. Количество комбинаций в первом $33*32*31*30*29*28*27*26*25 < 3^8*(11*2^5*2^5*10*29*25) = 3^8*2^{13}*(11*125*29*7)$ (мы заменили 26 на 27 и 31 на 32, увеличив число и тем самым вынесли больше за скобки). Во втором количество комбинаций $(2*3)^{17}$. Сокращаем и остаётся сравнивать $11*125*29*7$ и 2^4*3^9 . Подсчётом можно увидеть, что они равны 279125 и 314928 соответственно.*

Задание 5

В лесу Бельчонку досталась задача – соединить домики, стоящие на одной линии, телеграфными проводами, чтобы лесные жители могли общаться друг с другом. Провода проводятся только между соседними домами. Каждый дом должен быть соединён хотя бы с одним соседним, и при этом суммарно нужно потратить наименьшее количество провода. У Бельчонка есть ограниченное количество провода. Если его не хватит, программа должна выдать ответ NO.

Входные данные:

Сначала вводится N – количество столбов и M – длина имеющегося у Бельчонка провода. Потом вводятся координаты домов (различные неотрицательные целые числа) в количестве N .

Выходные данные: минимальная суммарная длина провода, либо NO, если минимальная длина меньше имеющихся у Бельчонка запасов.

Пример

№	Ввод	Вывод
1	4 10 17 2 4 20	5
2	7 20 44 15 1 20 4 41 3	11
3	4 20 96 1 3 5	NO

Ответ: Сначала сортируем массив координат, затем пользуемся методом динамического программирования. От двух точек ответом будет разность их координат ($S_2 = A_2 - A_1$), от более, чем двух – разность координат между последними столбами (ибо последний столб так или иначе придётся соединять с предпоследним) плюс $\min(S_{k-1}, S_{k-2})$.

Код в псевдоалгоритмическом языке:

Заводим числа N и M , массивы a и S

Считываем N и M

От $i = 1$ до N считываем в массив a

Сорт($a, a+n$); //возможна любая сортировка

От $i = 2$ до N : если $i = 2$, то $S[2] = a[2] - a[1]$

иначе $S[i] = a[i] - a[i-1] + \min(S[i-1], S[i-2])$

Если $S[N] \geq M$ то вывести $S[N]$

иначе NO