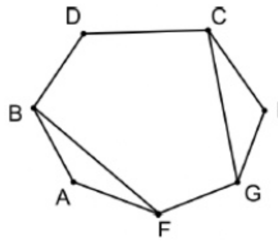
Вариант № 9169591

1. **Задание 1 №** [16030]

На рисунке слева изображена схема дорог Н-ского района, в таблице звёздочкой обозначено наличие дороги из одного населённого пункта в другой. Отсутствие звёздочки означает, что такой дороги нет.

	1	2	3	4	5	6	7
1					*	*	
2			*	*			*
3		*			*		*
4		*				*	
5	*		*			*	
6	*			*	*		
7		*	*				



Каждому населённому пункту на схеме соответствует его номер в таблице, но неизвестно, какой именно номер. Определите, какие номера населённых пунктов в таблице могут соответствовать населённым пунктам B и C на схеме. В ответе запишите эти два номера в возрастающем порядке без пробелов и знаков препинания.

Решение.

Заметим, что D - единственная вершина степени 2, которая связана с вершинами степени 3 — B и C, связаными с остальными вершинами степени 2. Значит, D соответствует П4. Таким образом, населённым пунктам B и C соответствуют пункты П2 и П6.

Ответ: 26.

Ответ: 26

2. **Задание 2 №** [14688]

Логическая функция *F* задаётся выражением $(x \lor y) \rightarrow (z \equiv x)$.

Дан частично заполненный фрагмент, содержащий **неповторяющиеся** строки таблицы истинности функции *F*.

Определите, какому столбцу таблицы истинности соответствует каждая из переменных *x*, *y*, *z*.

Переменная 1	Переменная 2	Переменная 3	Функция
???	???	???	*F*
	0	0	0
	0		0

В ответе напишите буквы *x*, *y*, *z* в том порядке, в котором идут соответствующие им столбцы (сначала – буква, соответствующая первому столбцу; затем – буква, соответствующая второму столбцу, и т. д.). Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

Пример. Пусть задано выражение $x \rightarrow y$, зависящее от двух переменных *x* и *y*, и фрагмент таблицы истинности:

Переменная 1	Переменная 2	Функция
???	???	*F*
0	1	0

Тогда первому столбцу соответствует переменная *y*, а второму столбцу соответствует переменная *x*. В ответе нужно написать: *yx*.

Решение.

Данная импликация принимает значение 0 тогда и только тогда, когда $\begin{cases} x+y=1, \\ x \neq z. \end{cases} \quad (*)$

Пусть x = 0, тогда y = z = 1. В первой строке нет двух единиц, значит, x = 1, и эта переменная находится в первом столбце. Тогда первая строка имеет вид 1 0 0.

Вторая строка должна отличаться от первой, поэтому она имеет вид 1 0 1. Рассмотрим два варианта:

x	y	z
1	0	0
1	0	1

x	z	y
1	0	0
1	0	1

Первый вариант не удовлетворяет системе (*), а второй удовлетворяет.

Ответ: xzy.

Приведем другое решение.

Составим таблицу истинности для выражения *(x ∨ y) → (z ≡ x)* и выпишем те наборы переменных, при которых данное выражение равно 0. В наборах переменные запишем в порядке *x, y, z*. Получим следующие наборы:

(0, 1, 1)

(1, 0, 0),

(1, 1, 0).

Сопоставим эти наборы с приведенным в задании фрагментом таблицы истинности.

Первая строка таблицы может соответствовать только набору (1, 0, 0), следовательно, первый столбец таблицы соответствует переменной *x*, и в первом столбце первой строки стоит 1.

Второй столбец таблицы может соответствовать только переменной *z*, поскольку переменная *y* принимает нулевое значение только в одном наборе. Тогда третий столбец соответствует переменной *y*.

Ответ: xzy

3. Задание 3 № 15620

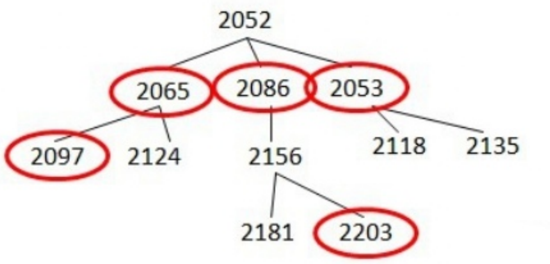
В фрагменте базы данных представлены сведения о родственных отношениях. На основании приведённых данных определите количество человек, у которых есть родной брат с разницей не более 5 лет.

Таблица 1			
ID	Фамилия_И. О.	Пол	Год рождения
2053	Сухорук К.К.	М	1975
2065	Лопухова В.А.	Ж	1980
2086	Зарецкий А.А.	М	1972
2097	Сухорук Е.К.	Ж	2004
2118	Ларина О.Д.	Ж	1996
2124	Сухорук И.К.	М	2001
2135	Кольцова Т.Х.	Ж	1995
2156	Рац А.П.	М	1993
2181	Сухорук Т.Н.	М	2015
2203	Сухорук П.И.	Ж	2018
2052	Гнатюк О.А.	М	1952

Таблица 2	
ID_Родителя	ID_Ребенка
2065	2097
2053	2118
2052	2065
2052	2086
2053	2135
2052	2053
2065	2124
2086	2156
2156	2181
2156	2203

Решение.

Используя данные таблиц, составим схему и выберем нужных людей, у которых есть брат с разницей не более 5 лет:



Ответ: 5.
Ответ: 5

4. Задание 4 № 15790

По каналу связи передаются сообщения, содержащие только семь букв: А, Б, Г, И, М, Р, Я. Для передачи используется двоичный код, удовлетворяющий условию Фано. Кодовые слова для некоторых букв известны: А — 010, Б — 011, Г — 100. Какое **наименьшее** количество двоичных знаков потребуется для кодирования слова МАГИЯ?

Примечание. Условие Фано означает, что ни одно кодовое слово не является началом другого кодового слова.

Решение.

Следующая буква должна кодироваться как 11, поскольку 10 мы взять не можем. 100 взять не можем из-за Г, значит, следующая буква должна быть закодирована кодом 101. Следующая буква должна кодироваться как 000, поскольку 00 взять не можем, иначе не останется кодовых слов для оставшейся буквы, которые удовлетворяют условию Фано. Значит, последняя буква будет кодироваться как 001. Тогда наименьшее количество двоичных знаков, которые потребуются для кодирования слова МАГИЯ равно 2 + 3 + 3 + 3 + 3 = 14.

Ответ: 14.

Примечание.

Заметим, что после кодирования всех букв, входящих в слово МАГИЯ, должен остаться хотя бы один свободный код для кодирования буквы Р, которая не входит в данное слово, но может передаваться по каналу связи. Проверить наличие свободного кода можно, построив дерево кодов, как показано в задаче 18553.

Ответ: 14

5. Задание 5 № 3421

У исполнителя, который работает с положительными однобайтовыми двоичными числами, две команды, которым присвоены номера:

1. сдвинь вправо
2. прибавь 4

Выполняя первую из них, исполнитель сдвигает число на один двоичный разряд вправо, а выполняя вторую, добавляет к нему 4. Исполнитель начал вычисления с числа 191 и выполнил цепочку команд 112112. Запишите результат в десятичной системе.

Решение.

При сдвиге вправо все биты числа в ячейке (регистре) сдвигаются на 1 бит вправо, в старший бит записывается нуль, а младший бит попадает в специальную ячейку – бит переноса, т. е. он теряется. Следовательно, если число чётное, то при сдвиге мы получаем число, в два раза меньше исходного; если число нечётное, в два раза меньше ближайшего меньшего чётного числа.

1: 191 перейдёт в 95,
1: 95 перейдёт в 47,
2: 47 перейдёт в 51,
1: 51 перейдёт в 25,
1: 25 перейдёт в 12,
2: 12 перейдёт в 16.

Ответ: 16.

Ответ: 16

6. Задание 6 № 13511

Запишите число, которое будет напечатано в результате выполнения следующей программы. Для Вашего удобства программа представлена на пяти языках программирования.

<table>
<tr><th>Бейсик</th><th>Python</th></tr>
<tr><td>DIM S, N AS INTEGER
S = 0
N = 0
WHILE 2*S*S < 10*S
    S = S + 1
    N = N + 2
WEND
PRINT N</td><td>s = 0
n = 0
while 2*s*s < 10*s:
    s = s + 1
    n = n + 2
print(n)</td></tr>
<tr><th>Паскаль</th><th>Алгоритмический язык</th></tr>
<tr><td>var s, n: integer;
begin
s := 0;
n := 0;
while 2*s*s < 10*s do
begin
    s := s + 1;
    n := n + 2;
end;
writeln(n)
end.</td><td>алг
нач
цел n, s
n := 0
s := 0
нц пока 2*s*s < 10*s
    s := s + 1
    n := n + 2
кц
вывод n
кон</td></tr>
<tr><th colspan="2">Си++</th></tr>
<tr><td colspan="2">#include <iostream>
using namespace std;
int main()
{ int s = 0, n = 0;
    while (2*s*s < 10*s) { s = s + 1; n = n + 2; }
    cout << n << endl;
    return 0;
}</td></tr>
</table>

Решение.

Цикл while выполняется до тех пор, пока истинно условие $2 \cdot s \cdot s < 10 \cdot s$. Заметим, что перед первым заходом в цикл переменная s равна 0, условие не выполняется. То есть переменная n будет равна 0.

Ответ: 0.
Ответ: 0

7. Задание 7 № 13355

Музыкальный фрагмент был оцифрован и записан в виде файла без использования сжатия данных. Получившийся файл был передан в город А по каналу связи за 15 секунд. Затем тот же музыкальный фрагмент был оцифрован повторно с разрешением в 2 раза выше и частотой дискретизации в 1,5 раза меньше, чем в первый раз. Сжатие данных не производилось. Полученный файл был передан в город Б; пропускная способность канала связи с городом Б в 2 раза выше, чем канала связи с городом А. Сколько секунд длилась передача файла в город Б? В ответе запишите только целое число, единицу измерения писать не нужно.

Решение.

Объём файла прямо пропорционален разрешению файла и его частоте дискретизации, следовательно, объём файла во втором случае в 2/1,5 = 4/3 раза больше. Длительность передачи обратно пропорциональна пропускной способности канала связи, откуда получаем, что длительность передачи файла во второй раз равна: $15 \cdot (1/2) \cdot (4/3) = 10$.

Ответ: 10.
Ответ: 10

8. Задание 8 № 10500

Шифр кодового замка представляет собой последовательность из пяти символов, каждый из которых является цифрой от 1 до 5. Сколько различных вариантов шифра можно задать, если известно, что цифра 1 встречается ровно три раза, а каждая из других допустимых цифр может встречаться в шифре любое количество раз или не встречаться совсем?

Решение.

Количество способов поставить три 1 на пять позиций — $C_5^3 = \frac{5!}{3!2!} = \frac{120}{12} = 10.$

После того, как определили позиции трёх 1, на оставшиеся две позиции можем поставить любое из четырёх чисел, это можно сделать $4^2 = 16$ способами.

Итого всего $10 \cdot 16 = 160$ кодов.

Ответ: 160

9. Задание 9 № 29657

Электронная таблица содержит результаты ежечасного измерения температуры воздуха на протяжении трёх месяцев. Определите, сколько раз за время наблюдений суточные колебания температуры (разность между максимальной и минимальной температурой в течение суток) превышали 17 градусов.

Задание 9

Решение.

Для поиска суточных колебаний температуры воспользуемся формулой =МАКС(**B2:Y2**)-МИН(**B2:Y2**) в ячейке Z2. Скопируем эту формулу во все ячейки диапазона Z3:Z92. Теперь в ячейке Z93 с помощью формулы =**СЧЁТЕСЛИ(Z2:Z92;">17")** найдём, сколько раз за время наблюдений суточные колебания температуры (разность между максимальной и минимальной температурой в течение суток) превышали 17 градусов. Ответ — 12.

Ответ: 12.

Ответ: 12

10. Задание 10 № 33755

Определите, сколько раз **в тексте** произведения Н. В. Гоголя «Нос» встречается существительное «шерсть» в любом падеже.

Задание 10

Решение.

Воспользуемся поисковыми средствами текстового редактора. В строке введём «шерст». Подсчитав общее количество результатов и исключив лишние, получаем ответ — 2.

Ответ: 2.

Примечание. Слово «шерстяную» не подходит, поскольку требуется найти существительное.

Ответ: 2

11. Задание 11 № 1904

В детскую игрушку «Набор юного шпиона» входят два одинаковых комплекта из четырех флажков различных цветов. Сколько различных тайных сообщений можно передать этими флажками, условившись менять выставленный флажок каждые пять минут и наблюдая за процессом 15 минут? Наблюдатель видит вынос первого флажка и две перемены флажка. При этом возможна смена флажка на флажок того же цвета.

Решение.

Каждое событие длится 5 минут, поэтому в сообщении длиной 15 минут будет N = 3 события. Одно событие по сути есть символ, а всё сообщение есть просто 3-буквенное слово.

Имеется 4 различных флажка, значит, 4 символа. Из M = 4 различных символов можно составить $Q = M^N$ слов длиной N = 3, т. е. $4^3 = 64$ слова.

Примечание. Цвета меняются по очереди, т. е. не нужно иметь три флажка одного цвета.

Ответ: 64.

Ответ: 64

12. Задание 12 № 10477

Исполнитель Редактор получает на вход строку цифр и преобразовывает её. Редактор может выполнять две команды, в обеих командах *v* и *w* обозначают цепочки цифр.

А) **заменить** (*v*, *w*).

Эта команда заменяет в строке первое слева вхождение цепочки *v* на цепочку *w*. Например, выполнение команды
заменить (555, 63)
преобразует строку 12555550 в строку 1263550.

Если в строке нет вхождений цепочки *v*, то выполнение команды **заменить** (*v*, *w*) не меняет эту строку.

Б) **нашлось** (*v*).

Эта команда проверяет, встречается ли цепочка *v* в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка исполнителя при этом не изменяется.

Цикл
ПОКА *условие*
последовательность команд
КОНЕЦ ПОКА
выполняется, пока условие истинно.
В конструкции
ЕСЛИ *условие*
ТО *команда1*
ИНАЧЕ *команда2*
КОНЕЦ ЕСЛИ
выполняется *команда1* (если условие истинно) или *команда2* (если условие ложно).

Какая строка получится в результате применения приведённой ниже программы к строке, состоящей из 1000 идущих подряд цифр 8? В ответе запишите полученную строку.

НАЧАЛО
ПОКА **нашлось** (999) ИЛИ **нашлось** (888)
ЕСЛИ **нашлось** (888)
ТО **заменить** (888, 9)
ИНАЧЕ **заменить** (999, 8)
КОНЕЦ ЕСЛИ
КОНЕЦ ПОКА
КОНЕЦ

Решение.

Данный алгоритм сначала заменит все триады из восьмёрок на девятки. Заметим, что 1000 восьмёрок образуют 333 триады и остаётся ещё одна восьмёрка. То есть получится 333 девятки и одна восьмёрка.

Затем алгоритм заменит 9 первых девяток на три восьмёрки, а затем заменит эти три восьмёрки обратно на одну девятку. То есть за четыре прохода цикла из начала строки убирается 8 девяток.

Таким образом, из строки, содержащей 333 девятки, уберется $8 \cdot 41=328$ девяток, и останется 5 девяток и одна восьмёрка. Три первых девятки будут заменены на одну восьмёрку, и останется строка 8998.

Ответ: 8998.

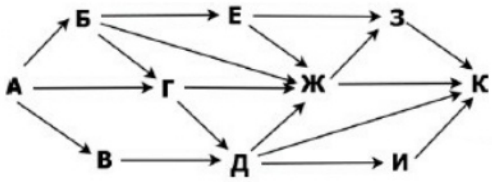
Примечание.

Рекомендуем сравнить эту задачу с задачей 10504. Исходные данные в этих задачах «симметричные» (строка из 1000 девяток или из 1000 восьмерок), но ответы получаются «несимметричные», что связано с порядком замены строк внутри цикла: в первую очередь заменяются строки из восьмерок.

Ответ: 8998

13. Задание 13 № 15631

На рисунке представлена схема дорог. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К, проходящих через город Г и НЕ проходящих через город З?



Решение.

Количество путей до города К равно количеству путей добраться в любой из тех городов, из которых есть дорога в Х.

При этом если путь должен не проходить через какой-то город, нужно просто не учитывать этот город при подсчёте сумм. А если город наоборот обязательно должен лежать на пути, тогда для городов, в которые из нужного города идут дороги, в суммах нужно брать только этот город.

С помощью этого наблюдения найдём последовательно количество путей до каждого из городов:

А = 1

Б = А = 1

Г = А + Б = 1 + 1 = 2 (т. к. проходить через город нужно Г обязательно)

Д = Г = 2

Ж = Г + Д = 4 (т. к. нельзя проходить через город З)

И = Д = 2

К = Ж + Д + И = 8

Ответ: 8.

Приведем другое решение.

Количество путей из города А в город К, проходящих через город Г, равно произведению количества путей из города А в город Г и количества путей из города Г в город К.

Из города А в город Г есть 2 пути: А—Г и А—Б—Г.

Найдем количество путей из города Г в город К (при этом Г - исходный пункт). По условию, эти пути не должны проходить через город З.

Г = 1

Д = Г = 1

Ж = Г + Д = 2

И = Д = 1

К = Ж + Д + И = 4.

Таким образом, количество путей из города А в город К, проходящих через город Г и не проходящих через город З, равно 2 · 4 = 8.

Ответ: 8

14. Задание 14 № 2309

Чему равно наименьшее основание позиционной системы счисления x, при котором $225_x = 405_y$?

Ответ записать в виде целого числа.

Решение.

Поскольку в левой и в правой частях есть цифра 5, оба основания больше 5, то есть перебор имеет смысл начинать с $x = x_{\min} = 6$.

Для каждого x вычисляем значение $225_x = 2 \cdot x^2 + 2x + 5 = N$ и решаем уравнение $N = 405_y = 4 \cdot y^2 + 5$, причем нас интересуют только натуральные $y > 5$.

Для $x = 6$ и $x = 7$ нужных решений нет, а для $x = 8$ получаем $N = 2 \cdot 8^2 + 2 \cdot 8 + 5 = 149 = 4 \cdot y^2 + 5$, так что $y = 6$.

Ответ: $x = 8$.

Ответ: 8

15. Задание 15 № 15634

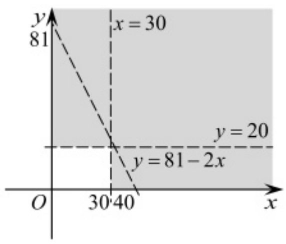
Для какого наименьшего целого неотрицательного числа A выражение

(y + 2x < A) ∨ (x > 30) ∨ (y > 20)

тождественно истинно, то есть принимает значение 1 при любых целых неотрицательных x и y?

Решение.

Решим задачу графически. Условия ($x > 30$) и ($y > 20$) задают множество, отмеченное на рисунке закрашенной областью. Чтобы исходное выражение было тождественно истинно для любых целых неотрицательных x и y, прямая ($y + 2x < A$) должна находиться правее незакрашенной области. Следовательно, она должна проходить через точки (30, 21) и (30,5, 20). Таким образом, наименьшее целое неотрицательное A равно 81.



Ответ: 81.

Ответ: 81

16. Задание 16 № 7459

Ниже на пяти языках программирования записан рекурсивный алгоритм F.

<table>
<tr><th>Бейсик</th><th>Python</th></tr>
<tr><td><pre>SUB F(n)
   PRINT n
   IF n < 5 THEN
      F(n + 1)
      F(n + 3)
   END IF
END SUB</pre></td><td><pre>def F(n):
   print(n)
   if n < 5:
      F(n + 1)
      F(n + 3)</pre></td></tr>
<tr><th>Паскаль</th><th>Алгоритмический язык</th></tr>
<tr><td><pre>procedure F(n: integer);
   begin
      writeln(n);
      if n < 5 then
         begin
            F(n + 1);
            F(n + 3)
         end
   end</pre></td><td><pre>алг F(цел n)
нач
вывод n, нс
если n < 5 то
   F(n + 1)
   F(n + 3)
все
кон</pre></td></tr>
<tr><th colspan="2">C++</th></tr>
<tr><td colspan="2"><pre>void F(int n)
{
   cout << n << endl;
   if (n < 5) {
      F(n + 1);
      F(n + 3);
   }
}</pre></td></tr>
</table>

Чему равна сумма всех чисел, напечатанных на экране при выполнении вызова F(1)?

Решение.

Первым действием процедура F(1) выведет число 1. Далее процедура F(1) вызовет процедуру F(n + 1), в результате выполнения которой на экране появится число n + 1 = 2. Процедура F(2) вызовет процедуру F(3), которая выведет на экран число 3 и вызовет процедуру F(4), которая выведет на экран число 4 и вызовет F(5), которая выведет на экран число 5.

После этого управление вернётся к процедуре F(4), которая начнёт выполнять следующий шаг своего алгоритма, т. е. обратиться к процедуре F(n + 3) = F(7). Процедура F(7) выведет на экран число 7. Далее управление вернётся к процедуре F(3). Рассуждая аналогично приходим к выводу, что процедура F(3) дополнительно выведет на экран число 6, процедура F(2) — 5.

Последним действием процедуры F(1) будет вызов процедуры F(n + 3) = F(4), которая выведет на экран числа 4, 5, 7.

Таким образом, на экране будут числа 1, 2, 3, 4, 5, 7, 6, 5, 4, 5, 7. Их сумма равна 49.

Ответ: 49.
Ответ: 49

17. Задание 17 № 27611

Рассматривается множество целых чисел, принадлежащих числовому отрезку [1813; 6861], которые делятся на 5 и не делятся на 6, 10, 15, 23. Найдите количество таких чисел и минимальное из них. В ответе запишите два целых числа без пробелов и других дополнительных символов: сначала количество, затем минимальное число.

Для выполнения этого задания можно написать программу или воспользоваться редактором электронных таблиц.

Решение.

Приведём решение данной задачи на языке Паскаль:

```
var count, min, i: integer;
begin
min := 20000;
count := 0;
for i := 1813 to 6861 do begin
if i mod 5 = 0 then
if i mod 6 <> 0 then
if i mod 10 <> 0 then
if i mod 15 <> 0 then
if i mod 23 <> 0 then begin
count := count + 1;
if i < min then
min := i;
end;
end;
writeln(count, min);
end.
```

Ответ: 3211825.

Ответ: 3211825

18. Задание 18 № 27683

Квадрат разлинован на $N \times N$ клеток ($1 < N < 17$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз — в соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. Перед каждым запуском Робота в каждой клетке квадрата лежит монета достоинством от 1 до 100. Посетив клетку, Робот забирает монету с собой; это также относится к начальной и конечной клетке маршрута Робота.

Задание 18

Откройте файл. Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой верхней клетки в правую нижнюю. В ответ запишите два числа друг за другом без разделительных знаков — сначала максимальную сумму, затем минимальную.

Исходные данные представляют собой электронную таблицу размером $N \times N$, каждая ячейка которой соответствует клетке квадрата.

Пример входных данных:

1	8	8	4
10	1	1	3
1	3	12	2
2	3	5	6

Для указанных входных данных ответом должна быть пара чисел 41 и 22.

Решение.

Для поиска максимального значения будем работать с областью A13:J22, так как при расчетах будем использовать исходные значения монет в каждой клетке. В ячейку A13 напишем значение =A1. Для каждой ячейки левого столбца это будет сумма всех ячеек выше от текущей. Внесем в ячейку A14 формулу =**A2+A13** и скопируем за маркер вниз до ячейки A22. Далее в ячейку B13 вставим формулу =B1+МАКС(**A13;B12**) и скопируем за маркер в ячейки B13:J22. Значение в ячейке J22 будет максимальной денежной суммой, которую сможет собрать Робот — 1044.

Аналогичным образом найдём значение минимальной денежной суммы. Вместо функции МАКС в диапазоне ячеек B13:J22 напишем функцию МИН. В таком случае значение в ячейке J22 будет минимальной денежной суммой, которую сможет собрать Робот — 448.

Ответ: 1044448.

Ответ: 1044448

19. Задание 19 № **27774**

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может **убрать из одной из куч один камень** или **уменьшить количество камней в куче в два раза** (если количество камней в куче нечётно, остаётся на 1 камень больше, чем убирается). Например, пусть в одной куче 6, а в другой 9 камней; такую позицию мы будем обозначать (6, 9). За один ход из позиции (6, 9) можно получить любую из четырёх позиций: (5, 9), (3, 9), (6, 8), (6, 5).

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не более 20. Победителем считается игрок, сделавший последний ход, то есть первым получивший позицию, в которой в кучах будет 20 или меньше камней.

В начальный момент в первой куче было 10 камней, во второй куче — S камней, $S > 10$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит, описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. В описание выигрышной стратегии не следует включать ходы играющего по ней игрока, которые не являются для него безусловно выигрышными, т.е не гарантирующие выигрыш независимо от игры противника.

Известно, что Ваня выиграл своим первым ходом после неудачного первого хода Пети. Укажите максимальное значение S, когда такая ситуация возможна.

Решение.

Такая ситуация возможна при $S = 40$. Если Петя уменьшит количество камней во второй куче в два раза, получится позиция (10, 20), из которой Ваня может получить позицию (10, 10) и выиграть. При $S > 40$ никакой первый ход Пети не создаст ситуацию, в которой Ваня может сразу выиграть.

Ответ: 40.

Примечание.

Рекомендуем сравнить эту задачу с задачей 27771.

Ответ: 40

20. Задание 20 № 27775

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может **убрать из одной из куч один камень** или **уменьшить количество камней в куче в два раза** (если количество камней в куче нечётно, остаётся на 1 камень больше, чем убирается). Например, пусть в одной куче 6, а в другой 9 камней; такую позицию мы будем обозначать (6, 9). За один ход из позиции (6, 9) можно получить любую из четырёх позиций: (5, 9), (3, 9), (6, 8), (6, 5).

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не более 20. Победителем считается игрок, сделавший последний ход, то есть первым получивший позицию, в которой в кучах будет 20 или меньше камней.

В начальный момент в первой куче было 10 камней, во второй куче — S камней, $S > 10$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит, описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. В описание выигрышной стратегии не следует включать ходы играющего по ней игрока, которые не являются для него безусловно выигрышными, т. е не гарантирующие выигрыш независимо от игры противника.

Найдите пять таких значений S, при которых у Пети есть выигрышная стратегия, причём одновременно выполняются два условия:

— Петя не может выиграть за один ход;

— Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня.

Найденные значения запишите в ответе в порядке возрастания без разделительных знаков.

Решение.

Возможные значения S: 42, 41, 31, 23, 22. В этих случаях Петя, очевидно, не может выиграть первым ходом. Однако при $S = 31$ Петя может получить позицию (5, 31), при $S = 23$ — позицию (9, 23), при $S = 22$ — позицию (10, 21), а при $S = 41$ и $S = 42$ — позицию (10, 21). В первом случае после хода Вани возникнет одна из позиций (4, 31), (3, 31), (5, 30), (5, 16), во втором случае — одна из позиций (8, 23), (5, 23), (9, 22), (9, 12), в третьем случае — одна из позиций (9, 21), (10, 20), (5, 21), (10, 11), в четвёртом и пятом случаях — одна из позиций (9, 21), (10, 20), (5, 21), (10, 11). В любой из перечисленных позиций Петя может выиграть, уменьшив вдвое количество камней в большей куче.

Таким образом, ответ — 2223314142.

Ответ: 2223314142.

Ответ: 2223314142

21. Задание 21 № 27776

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может **убрать из одной из куч один камень** или **уменьшить количество камней в куче в два раза** (если количество камней в куче нечётно, остаётся на 1 камень больше, чем убирается). Например, пусть в одной куче 6, а в другой 9 камней; такую позицию мы будем обозначать (6, 9). За один ход из позиции (6, 9) можно получить любую из четырёх позиций: (5, 9), (3, 9), (6, 8), (6, 5).

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не более 20. Победителем считается игрок, сделавший последний ход, то есть первым получивший позицию, в которой в кучах будет 20 или меньше камней.

В начальный момент в первой куче было 10 камней, во второй куче — S камней, $S > 10$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока — значит, описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. В описание выигрышной стратегии не следует включать ходы играющего по ней игрока, которые не являются для него безусловно выигрышными, т.е не гарантирующие выигрыш независимо от игры противника.

Найдите максимальное значение S, при котором одновременно выполняются два условия:

— у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;

— у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Решение.

Возможное значение S: 24. После первого хода Пети возможны позиции (9, 24), (5, 24), (10, 23), (10, 12). В позициях (5, 24) и (10, 12) Ваня может выиграть первым ходом, уменьшив вдвое количество камней во второй куче. Из позиций (9, 24) и (10, 23) Ваня может получить позицию (9, 23), в этом случае после хода Пети возникнет одна из позиций (8, 23), (5, 23), (9, 22), (9, 12). В любой из перечисленных позиций Ваня может выиграть, уменьшив вдвое количество камней в большей куче.

Таким образом, ответ — 24.

Ответ: 24.

Примечание.

Заметим, что при уменьшении количества камней в два раза в случае нечетного количества камней остается на один камень больше, чем убирается. Поэтому значение S = 32 не подходит: в этом случае Петя может получить позицию (9, 32). Возможные ходы Вани:

(9, 16) — тогда Петя выигрывает ходом (9, 8).

(9, 31) — тогда Петя делает ход (8, 31).

(8, 32) — тогда Петя делает ход (7, 32).

(5, 32) — тогда Петя делает ход (5, 31).

Ни в одой из получившихся позиций Ваня не сможет выиграть следующим ходом.

Ответ: 24

22. Задание 22 № 13496

Ниже на пяти языках программирования записан алгоритм. Получив на вход натуральное число x, этот алгоритм печатает число S. Укажите такое наименьшее число x, при вводе которого алгоритм печатает шестизначное число.

Бейсик	Python

DIM X,S,D,R AS LONG
INPUT X
S = X
R = 0
WHILE X > 0
  D = X MOD 2
  R = 10*R + D
   X = X \ 2
WEND
S = R + S
PRINT S
	x = int(input())
	S = x;
	R = 0
	while x > 0:
	   d = x % 2
	   R = 10*R + d
	  x=x // 2
	S = R + S
	print(S)

Паскаль	**Алгоритмический язык**

var x,d,R,S: longint;
begin
    readln(x);
    S := x;
    R := 0;
    while x > 0 do
    begin
      d := x mod 2;
      R := 10*R + d;
      x := x div 2;
    end;
    S := R + S;
    writeln(S);
end.
	алг
	нач
	    цел x, d, R, S
	    ввод x
	    S := x
	    R := 0
	    нц пока x > 0
	       d := mod(x, 2)
	       R := 10*R + d
	       x := div(x, 2)
	    кц
	    S := R + S
	   вывод S
	кон

Си++

```
#include <iostream>
using namespace std;
int main()
{
    long x,d,R,S;
    cin >> x;
    S = x;
    R = 0;
    while (x > 0){
        d = x % 2;
        R = 10*R + d;
        x = x / 2;
      }
    S = R + S;
    cout << S << endl;
   return 0;
}
```

Решение.

Для того, чтобы итоговое число было шестизначным, нужно, чтобы цикл выполнился 6 раз. Тогда нужно минимальное число, которое 6 раз поделится на 2 (при шестом делении на 2 результат будет равен 0). Известно, что число больше 32 (т.к. это минимальное число, которое 6 раз делится на 2), а следующее за ним подходит.

Ответ: 33.

Ответ: 33

23. Задание 23 № 3303

У исполнителя Калькулятор две команды:

1. прибавь 2
2. прибавь 3.

Первая из них увеличивает число на экране на 2, вторая — на 3. Сколько различных чисел можно получить из числа 2 с помощью программы, которая содержит ровно 10 команд?
Решение.
Для сложения справедлив переместительный (коммутативный) закон, значит, порядок команд в программе не имеет значения.

Каждой программе соответствует одно число, поэтому посчитав количество возможных программ (с точностью до перестановки), найдём количество различных чисел.

Если в программе n команд 1, тогда в ней будет 10-n команд 2. n изменяется от 0 до 10. Всего 11 программ, следовательно, 11 чисел.

Ответ: 11.
Ответ: 11

24. Задание 24 № 27690

Текстовый файл состоит не более чем из 10^6 символов *A*, *B* и *C*. Определите максимальное количество идущих подряд символов, среди которых каждые два соседних различны.

Для выполнения этого задания следует написать программу. Ниже приведён файл, который необходимо обработать с помощью данного алгоритма.

Задание 24

Решение.

Для решения данной задачи будем посимвольно считывать текстовый файл. Объявим переменные: maxLen — максимальная длина последовательности, curLen — временное хранение длины последовательности, i — переменная для перебора всех символов, s — строка для работы с символами из файла. Алгоритм будет сравнивать значение текущего символа со значением предыдущего, если символы будут удовлетворять нужным условиям, то значение счетчика будет увеличиваться на 1.

Приведём решение данной задачи на языке Pascal.

```
var maxLen, curLen, i: integer;
    s: string;
begin
  assign(input, '24.txt');
  readln(s);
  maxLen := 1;
  curLen := 1;
  for i:=2 to Length(s) do
    if s[i] <> s[i-1] then begin
      curLen := curLen + 1;
      if curLen > maxLen then
        maxLen := curLen;
    end
    else
      curLen := 1;
  writeln(maxLen);
end.
```

В результате работы данного алгоритма при вводе данных из файла в условии получаем ответ — 42.

Ответ: 42.
Ответ: 42

25. Задание 25 № 27854

Напишите программу, которая ищет среди целых чисел, принадлежащих числовому отрезку [110203; 110245], числа, имеющие ровно четыре различных чётных натуральных делителя (при этом количество нечётных делителей может быть любым). Для каждого найденного числа запишите эти четыре делителя в четыре соседних столбца на экране с новой строки. Делители в строке должны следовать в порядке возрастания.

Например, в диапазоне [2; 16] ровно четыре чётных различных натуральных делителя имеют числа 12 и 16, поэтому для этого диапазона вывод на экране должна содержать следующие значения:

2 4 6 12
2 4 8 16

Ответ:

Решение.

Решим задачу перебором. Будем проверять количество делителей каждого числа из диапазона, если их количество равно четырём — записываем их в двумерный массив *d*. После этого выводим эти делители на экран в новой строке.

Приведём решение на языке Pascal.

```
var
numDel, i, j: longint;
d2: array[1..4] of longint;
begin
for i := 110203 to 110245 do begin
numDel := 0;
for j := 1 to i do begin
if (i mod j = 0) and (j mod 2 = 0) then begin
numDel := numDel + 1;
if numDel > 4 then break;
d2[numDel] := j;
end;
end;
if numDel = 4 then writeln(d2[1], ' ', d2[2], ' ', d2[3], ' ', d2[4]);
end;
end.
```

В результате работы программа должна вывести следующее:

2 4 55102 110204
2 14 15746 110222
2 6 36742 110226
2 22 10022 110242

Ответ: 2&4&55102&110204&2&14&15746&110222&2&6&36742&110226&2&22&10022&110242

26. Задание 26 № 27885

Системный администратор раз в неделю создаёт архив пользовательских файлов. Однако объём диска, куда он помещает архив, может быть меньше, чем суммарный объём архивируемых файлов. Известно, какой объём занимает файл каждого пользователя.

По заданной информации об объёме файлов пользователей и свободном объёме на архивном диске определите максимальное число пользователей, чьи файлы можно сохранить в архиве, а также максимальный размер имеющегося файла, который может быть сохранён в архиве, при условии, что сохранены файлы максимально возможного числа пользователей.

Входные данные.

Задание 26

В первой строке входного файла находятся два числа: S — размер свободного места на диске (натуральное число, не превышающее 10 000) и N — количество пользователей (натуральное число, не превышающее 3000). В следующих N строках находятся значения объёмов файлов каждого пользователя (все числа натуральные, не превышающие 100), каждое в отдельной строке.

Запишите в ответе два числа: сначала наибольшее число пользователей, чьи файлы могут быть помещены в архив, затем максимальный размер имеющегося файла, который может быть сохранён в архиве, при условии, что сохранены файлы максимально возможного числа пользователей.

Пример входного файла:

100 4
80
30
50
40

При таких исходных данных можно сохранить файлы максимум двух пользователей. Возможные объёмы этих двух файлов 30 и 40, 30 и 50 или 40 и 50. Наибольший объём файла из перечисленных пар — 50, поэтому ответ для приведённого примера:

2 50

Ответ:

Решение.

Сначала считаем в массив данные из файла. После этого отсортируем массив в порядке возрастания. Таким образом, последовательно складывая элементы массива с начала и сравнивая сумму с размером свободного места на диске получим максимальное количество пользователей, чьи файлы могут поместиться на диске. Далее, вычитая из найденной суммы наибольший файл в текущей последовательности, будем пробовать прибавлять файлы с большим весом. Если такой файл будет найден, то заменяем значение наибольшего файла, который возможно поместить на диск.

Приведём решение на языке Pascal.

```
var
i, j, t: integer;
a: array [1..3000] of integer;
s: integer;
n: integer;
sum: integer;
maxi: integer;
f: text;
begin
assign(f,'C:\27885.txt');
reset(f);
readln(f, s, n);
for i := 1 to n do readln(f, a[i]);
for i := 1 to n do
for j := i + 1 to n do
if a[i] > a[j] then begin
t := a[i];
a[i] := a[j];
a[j] := t;
end;
sum := 0;
maxi := 1;
for i := 1 to n do
if sum + a[i] <= s then begin
sum := sum + a[i];
maxi := i;
end;
t := a[maxi];
for i := maxi to n do
if ((sum - t) + a[i]) <= s then begin
sum := sum - t + a[i];
t := a[i];
end;
writeln(maxi, ' ', t);
end.
```

Ответ: 650 27.

Примечание. Путь к файлу необходимо указать согласно расположению файла на Вашем компьютере.

Ответ: 650&27

27. Задание 27 № 33529

Набор данных состоит из нечётного количества пар натуральных чисел. Необходимо выбрать из каждой пары ровно одно число так, чтобы чётность суммы выбранных чисел совпадала с чётностью большинства выбранных чисел и при этом сумма выбранных чисел была как можно больше. Определите максимальную сумму, которую можно получить при таком выборе. Гарантируется, что удовлетворяющий условиям выбор возможен.

Входные данные.

Файл А

Файл B

Первая строка входного файла содержит число N — общее количество пар в наборе. Каждая из следующих N строк содержит два натуральных числа, не превышающих 10 000.

Пример входного файла:

5
15 8
5 11
6 3
7 2
9 14

Для указанных данных надо выбрать числа 15, 11, 6, 7 и 14. Большинство из них нечётны, сумма выбранных чисел равна 53 и тоже нечётна. В ответе надо записать число 53.

Вам даны два входных файла (A и B), каждый из которых имеет описанную выше структуру. В ответе укажите два числа: сначала значение искомой суммы для файла A, затем для файла B.

Предупреждение: для обработки файла B не следует использовать переборный алгоритм, вычисляющий сумму для всех возможных вариантов, поскольку написанная по такому алгоритму программа будет выполняться слишком долго.

Ответ:

Решение.

Последовательно считывая данные из файла, будем прибавлять к сумме значение максимального числа в паре, при этом, если число чётное, будем увеличивать значение переменной count0 на единицу, если нечётное — увеличивать значение переменной count1 на единицу. Поскольку может возникнуть ситуация, когда, например, получившаяся сумма будет чётной, а количество чётных чисел будет меньше количества нечётных чисел и будет отличаться от количества нечётных чисел на единицу, будем находить две минимальных разницы для ситуации, когда будет убираться два чётных числа (переменные dif3 и dif4), и две минимальных разницы для ситуации, когда будет убираться два нечётных числа (переменные dif1 и dif2).

Приведём решение задачи на языке Pascal.

```
var
x, y, count0, count1: integer;
n: integer;
sum: integer;
dif1, dif2, dif3, dif4: integer;
f: text;
begin
    assign(f,'C:\27-B.txt');
    reset(f);
    readln(f, n);
    sum := 0;
    dif1 := 20001;
    dif2 := 20001;
    dif3 := 20001;
    dif4 := 20001;
    count0 := 0;
    count1 := 0;
    while not eof(f) do begin
        readln(f, x, y);
        if x > y then begin
            sum := sum + x;
            if x mod 2 = 0 then count0 := count0 + 1
            else count1 := count1 + 1;
            if x mod 2 <> y mod 2 then begin
                if (x - y < dif1) and (x mod 2 <> 0) then begin
                    dif2 := dif1;
                    dif1 := x - y;
```

```
            end
            else if (x - y < dif2) and (x mod 2 <> 0) then
                dif2 := x - y;
            if (x - y < dif3) and (x mod 2 = 0) then begin
                dif4 := dif3;
                dif3 := x - y;
            end
            else if (x - y < dif4) and (x mod 2 = 0) then
                dif4 := x - y;
        end;
    end
    else begin
        if y mod 2 = 0 then count0 := count0 + 1
        else count1 := count1 + 1;
        sum := sum + y;
        if x mod 2 <> y mod 2 then begin
            if (y - x < dif1) and (y mod 2 <> 0) then begin
                dif2 := dif1;
                dif1 := y - x;
            end
            else if (y - x < dif2) and (y mod 2 <> 0) then
                dif2 := y - x;
            if (y - x < dif3) and (y mod 2 = 0) then begin
                dif4 := dif3;
                dif3 := y - x;
            end
            else if (y - x < dif4) and (y mod 2 = 0) then
                dif4 := y - x;
        end;
    end;
end;
if (count1 > count0) then begin
    if sum mod 2 = 1 then
        writeln(sum)
    else if dif3 <= dif1 then
        writeln(sum - dif3)
    else if (count1 - count0) <> 1 then
        writeln(sum - dif1)
    else if (dif1 + dif2) < dif3 then
        writeln(sum - dif1 - dif2)
    else writeln(sum - dif3)
end
else begin
    if sum mod 2 = 0 then
        writeln(sum)
    else if dif1 <= dif3 then
        writeln(sum - dif1)
    else if (count0 - count1) <> 1 then
        writeln(sum - dif3)
    else if (dif3 + dif4) < dif1 then
        writeln(sum - dif3 - dif4)
    else writeln(sum - dif1)
end;
end.
```

В результате работы данного алгоритма при вводе данных из файла A ответ — 121184, из файла B — 36898658.

Примечание. Путь к файлу необходимо указать согласно расположению файла на Вашем компьютере.
Ответ: 121184&36898658