

ВАРИАНТ 3

Задание 3.1

Дано натуральное число $N > 10$, в десятичной записи которого нет нулей. Необходимо определить минимальное двузначное число, которое можно увидеть в десятичной записи N . Например, для $N=1984$ нужно получить результат 19, а для $N = 271\ 828$ – результат 18. Для решения этой задачи ученик написал программу, но, к сожалению, его программа неправильная.

Ниже эта программа для Вашего удобства приведена на пяти языках программирования.

Бейсик	Python
<pre>DIM N, K AS INTEGER INPUT N K = 100 WHILE N > 100 IF N MOD 100 < K THEN K = N MOD 100 END IF N = N \ 100 WEND PRINT K END</pre>	<pre>n = int(input()) k = 100 while n > 100: if n%100 < k: k = n%100 n = n // 100 print(k)</pre>
Алгоритмический язык	Паскаль
<pre><u>алг</u> <u>нач</u> <u>цел</u> n, k <u>ввод</u> n k := 100 <u>нц пока</u> n > 100 <u>если</u> mod(n,100) < k <u>то</u> k:=mod(n,100) <u>все</u> n:=div(n,100) <u>кц</u> <u>вывод</u> k <u>кон</u></pre>	<pre>var n, k: integer; begin read(n); k := 100; while n > 100 do begin if n mod 100 < k then k := n mod 100; n := n div 100; end; writeln(k) end.</pre>
Си	
<pre>#include <stdio.h> int main() { int n, k; scanf("%d", &n); k = 100; while (n > 100) { if (n%100 < k) k = n%100; n = n/100; } printf("%d", k); return 0; }</pre>	

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе $N=6789$.
 2. Приведите два возможных значения N , при вводе которых программа выведет верный ответ. Укажите этот ответ.
 3. Найдите в программе все ошибки (их может быть одна или несколько). Для каждой ошибки выпишите строку, в которой она допущена, и приведите эту же строку в исправленном виде.
- Достаточно указать ошибки и способ их исправления для одного языка программирования. Обратите внимание: Вам нужно исправить приведённую программу, а не написать свою. Вы можете только заменять ошибочные строки, но не можете удалять строки или добавлять новые. Заменять следует только ошибочные строки: за исправления, внесённые в строки, не содержащие ошибок, баллы будут снижаться.

Содержание верного ответа

(допускаются иные формулировки ответа, не искажающие его смысла)

1. При вводе $N=6789$ программа выведет число 89.
2. Примеры чисел, при вводе которых программа выводит верный ответ: 1815 (ответ 15), 412 756 (ответ 27).

Комментарии для экспертов. Из-за ошибки в программе не все пары соседних цифр рассматриваются как возможные двузначные числа. Обрабатываются только те пары, правее которых расположено чётное число цифр, кроме пары в начале числа. Например, в числе 87 654 321 рассматриваются только пары 21, 43, 65. Сравнение в программе организовано верно, поэтому верный ответ выдаётся в том случае, когда пара цифр, образующая самое маленькое число, расположена на местах, попадающих в обработку.

3. Программа содержит две ошибки:

- 1) неверное условие цикла;
- 2) неверное изменение величины N .

Пример исправления для языка Паскаль

Первая ошибка:

```
while n > 100 do begin
```

Исправленная строка:

```
while n > 10 do begin
```

Вместо сравнения с 10 возможно сравнения с 0 или любым другим числом, меньшим 10. Вместо строгого сравнения возможно нестрогое. Возможно также сравнение с числом 11, но в этом случае обязательно нестрогое ($n \geq 11$).

Вторая ошибка:

```
n := n div 100;
```

Исправленная строка:

```
n := n div 10;
```

В программах на других языках ошибочные строки и их исправления аналогичны.

Незначительной опiskeй, не влияющей на оценку, следует считать отсутствие знаков и служебных слов после содержательной части исправления.

Задание 3.2

Дан массив, содержащий 2016 положительных целых чисел, не превышающих 1000. Необходимо найти и вывести максимальный из тех элементов этого массива, шестнадцатеричная запись которых заканчивается символом В. Если таких чисел в массиве нет, ответ считается равным нулю. Исходные данные объявлены так, как показано ниже. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать часть из описанных переменных.

Бейсик	Python
CONST N=2016 DIM A(N) AS INTEGER DIM I, M, K AS INTEGER FOR I = 1 TO N INPUT A(I) NEXT I ... END	# допускается также использование # целочисленных переменных m, k a = [] N = 2016 for i in range(0, N): a.append(int(input())) ...
Алгоритмический язык	Паскаль
<u>алг</u> <u>нач</u> <u>цел</u> N=2016 <u>целтаб</u> a[1:N] <u>цел</u> i, m, k <u>нц для</u> i <u>от</u> 1 <u>до</u> N <u>ввод</u> a[i] <u>кц</u> ... <u>кон</u>	const N=2016; var a: array [1..N] of integer; i, m, k: integer; begin for i:=1 to N do readln(a[i]); ... end.
Си	
#include <stdio.h> #define N 2016 int main(){ int a[N]; int i, m, k; for (i=0; i<N; i++) scanf("%d", &a[i]); ... return 0; }	

В качестве ответа Вам необходимо привести фрагмент программы, который должен находиться на месте многоточия. Вы можете записать решение также на другом языке программирования (укажите название и версию языка программирования). В этом случае Вы должны использовать те же самые исходные данные и переменные, какие были предложены в условии.

Содержание верного ответа

(допускаются иные формулировки ответа, не искажающие его смысла)

Последний символ записи числа в позиционной системе счисления показывает остаток от деления этого числа на основание системы. Чтобы шестнадцатеричная запись числа

заканчивалась символом В, число должно при делении на 16 давать остаток 11.

Для решения задачи необходимо просмотреть все числа в массиве и выбрать максимальное среди тех, которые дают остаток 11 при делении на 16

Пример правильной программы на языке Паскаль

```
m:=0;
for i:=1 to N do begin
  if (a[i] mod 16 = 11) and (a[i]>m)
    then m := a[i];
end;
writeln(m)
```

Задание 3.3

Два игрока, Паша и Валя, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Паша. За один ход игрок может добавить в кучу **один** камень или увеличить количество камней в куче в **три** раза. Например, имея кучу из 15 камней, за один ход можно получить кучу из 16 или 45 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней. Игра завершается в тот момент, когда количество камней в куче становится не менее 36. Если при этом в куче оказалось не более 98 камней, то победителем считается игрок, сделавший последний ход. В противном случае победителем становится его противник. Например, если в куче было 33 камня и Паша утроит количество камней в куче, то игра закончится и победителем будет Валя. В начальный момент в куче было S камней, $1 \leq S \leq 35$.

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. Выполните следующие задания.

- а) При каких значениях числа S Паша может выиграть в один ход?
Укажите все такие значения и соответствующие ходы Паши.
- б) У кого из игроков есть выигрышная стратегия при $S = 34; 33; 32$?
Опишите выигрышные стратегии для этих случаев.
2. У кого из игроков есть выигрышная стратегия при $S = 11; 10$? Опишите соответствующие выигрышные стратегии.
3. У кого из игроков есть выигрышная стратегия при $S = 9$? Постройте дерево всех партий, возможных при этой выигрышной стратегии (в виде рисунка или таблицы). На рёбрах дерева указывайте, кто делает ход, в узлах – количество камней в позиции.

Содержание верного ответа

(допускаются иные формулировки ответа, не искажающие его смысла)

1. а) Паша может выиграть, если $S = 35$ или $S = 12; 13; \dots; 32$. При $S = 35$ первым ходом нужно добавить в кучу 1 камень, при остальных указанных значениях S нужно утроить количество камней.

б) При $S = 32$ Паша выигрывает в один ход, утраивая количество камней (см. п. а). При $S = 33$ или 34 утраивать количество камней не имеет смысла, так как после такого хода выигрывает противник. Поэтому можно считать, что единственный возможный ход – это добавление в кучу одного камня.

При $S = 34$ после такого хода Паши в куче станет 35 камней. В этой позиции ходящий (т. е. Валя) выигрывает (см. п. а)). Т. е. при $S = 34$ Паша (игрок, который должен ходить первым) проигрывает. Выигрышная стратегия есть у Вали.

При $S = 33$ после того, как Паша своим первым ходом добавит один камень, в куче станет 34 камня. В этой позиции ходящий (т. е. Валя) проигрывает (см. выше). Т. е. при $S = 33$ Паша (игрок, который должен ходить первым) выигрывает. Выигрышная стратегия есть у Паши.

Комментарии для экспертов. Скорее всего, решение экзаменуемого будет не столь подробным. Это не является ошибкой. Ученик может, например, нарисовать деревья всех возможных партий для указанных значений S . Другая возможность – (1) указать на то, что при $S = 33$ и 34 утраивать кучу смысла не имеет, и (2) последовательно сводить случай $S = 34$ к случаю $S = 35$, а случай $S = 33$ – к случаю $S = 34$.

2. При $S = 11$ после первого хода Паши в куче будет либо 12 камней, либо 33 камня. В обоих случаях выигрышная стратегия есть у игрока, который должен ходить, теперь это Валя. Случай $S = 12$ рассмотрен в задании 1(а), а случай $S = 33$ – в задании 1(б). Поэтому выигрышная стратегия есть у Вали.

При $S = 10$ выигрышная стратегия есть у Паши. Ему нужно первым ходом добавить 1 камень и получить кучу из 11 камней. Как показано выше, в этой ситуации выигрышная стратегия есть у игрока, который НЕ должен ходить, т. е. у Паши.

3. При $S = 9$ выигрышная стратегия есть у Вали. После первого хода Паши в куче может стать либо 10 камней, либо 27 камней. В обоих этих позициях выигрывает игрок, который будет делать ход (теперь это Валя). Случай $S = 10$ рассмотрен в п. 2, случай $S = 27$ рассмотрен в п. 1(а).

В таблице изображено дерево возможных партий при описанной стратегии Вали. Заключительные позиции (в них выигрывает Валя) подчёркнуты. На рисунке это же дерево изображено в графическом виде (оба способа изображения дерева допустимы).

		Положения после очередных ходов				
Исх. положени е. 9	1-й ход Паши (все ходы) 9+1=10 9*3=27	1-й ход Вали (только ход по стратегии)	2-й ход Паши (все ходы)	2-й ход Вали (только ход по стратегии)	3-й ход Паши (все ходы)	3-й ход Вали (только ход по стратегии)
		10+1=11	11*3=33	33+1=34	34+1=35	35+1=36
					34*3=102	
				11+1=12	12*3=36	
		27*3=81				

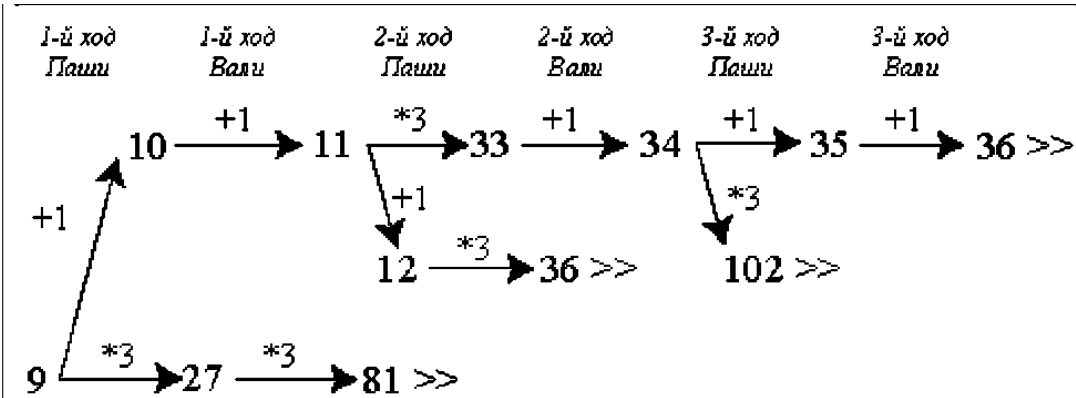


Рис.1. Дерево всех партий, возможных при Валиной стратегии.
 Знаком >> обозначены позиции, в которых партия заканчивается

Задание 3.4

Дан набор из N целых положительных чисел. Необходимо выбрать из набора произвольное количество чисел так, чтобы их сумма была как можно больше и при этом не делилась на 6. В ответе нужно указать количество выбранных чисел и их сумму, сами числа выводить не надо. Если получить нужную сумму невозможно, считается, что выбрано 0 чисел и их сумма равна 0. Напишите эффективную по времени и по памяти программу для решения этой задачи.

Программа считается эффективной по времени, если при увеличении количества исходных чисел N в k раз время работы программы увеличивается не более чем в k раз.

Программа считается эффективной по памяти, если память, необходимая для хранения всех переменных программы, не превышает 1 килобайт и не увеличивается с ростом N .

Максимальная оценка за правильную (не содержащую синтаксических ошибок и дающую правильный ответ при любых допустимых входных данных) программу, эффективную по времени и по памяти, – 4 балла. Максимальная оценка за правильную программу, эффективную только по времени или только по памяти, – 3 балла.

Максимальная оценка за правильную программу, не удовлетворяющую требованиям эффективности, – 2 балла.

Вы можете сдать одну или **две** программы решения задачи. Если Вы сдадите две программы, каждая из них будет оцениваться независимо от другой, итоговой станет **большая** из двух оценок.

Перед текстом программы кратко опишите алгоритм решения. Укажите использованный язык программирования и его версию.

Описание входных и выходных данных

В первой строке входных данных задаётся количество чисел N ($1 \leq N \leq 1000$).

В каждой из последующих N строк записано одно натуральное число, не превышающее 10 000.

Пример входных данных:

```
3
1
2
3
```

В результате работы программа должна вывести два числа: сначала количество выбранных чисел, затем их сумму.

Пример выходных данных для приведённого выше примера входных данных:

```
2 5
```

В данном случае из предложенного набора нужно выбрать два числа (2 и 3), их сумма равна 5.

Содержание верного ответа

(использована версия PascalABC)

(допускаются иные формулировки ответа, не искажающие его смысла)

Если сумма всех данных чисел не кратна 6, нужно просто взять все числа. Если сумма кратна 6, нужно удалить из нее минимально возможный элемент – наименьшее из данных чисел, не кратное 6. Если таких чисел нет (все числа в наборе кратны 6), то получить требуемую сумму невозможно, в этом случае по условию задачи ответ считается равным нулю. Программа должна прочитать все числа, не сохраняя их, подсчитать общую сумму и определить наименьшее число, не кратное 6, а далее действовать по описанным выше правилам.

Ниже приведена реализующая этот алгоритм программа на языке Паскаль (использована версия Pascal ABC).

Пример правильной и эффективной программы на языке Паскаль

```
const d=6; amax = 10000;    {делитель}
{максимально возможное число}
var
N: integer; a: integer; s: integer; mn: integer; k: integer; i: integer; {количество чисел}
{очередное число}
{сумма}
{минимальное число, не кратное d}
{количество выбранных чисел}
begin readln(N); s := 0; mn := amax+1; for i:=1 to N do begin readln(a); s := s+a;
if (a mod d <> 0) and (a < mn) then mn := a;
end;
if s mod d <> 0 then k := N
else if mn <= amax then begin
k := N-1;
s := s - mn; end else begin
k := 0;
s := 0; end;
writeln(k, ' ', s); end.
```