

ЗАДАНИЕ ПО ИНФОРМАТИКЕ ВАРИАНТ 73991 для 9 класса

Для заданий 1-4 требуется разработать алгоритмы на языке блок-схем,
псевдокоде или естественном языке

1. Известно, что четырнадцатизначное число $A = \overline{20x1zux2022xu4}$ делится нацело на число $\overline{y2x}$. Составьте алгоритм для нахождения всех возможных троек цифр (x, y, z) .

Решение. Поскольку число A заканчивается цифрой 4, то x не может быть 0 и 5. y и z могут быть любыми. Исключить из рассмотрения только 0 очень просто, исключить из рассмотрения и 0, и 5 чуть сложнее. Сделать это можно несколькими способами.

1. Вставить условие $x \neq 5$ в цикл для x от 1 до 9.
2. Написать два цикла – для x от 1 до 4 и для x от 6 до 9.
3. Написать цикл для x от 1 до 4, и на каждом шаге цикла в качестве цифры брать значения x и $x + 5$.

Возникает вопрос целесообразности этих модификаций. Исключение из рассмотрения одной цифры уменьшает время работы программы на 10%. Исключение из рассмотрения ещё одной цифры уменьшает время работы программы ещё на 10%, но увеличивает объём самой программы. При длительном времени работы программы уменьшение времени на 20%, несомненно, целесообразно. В данном случае, наверное, оптимальным будет выбрать беззатратный вариант с исключением только цифры 0.

В общем, надо тем или иным образом организовать цикл для перебора значений x , внутри него цикл по y от 0 до 9 с шагом 1 и цикл по z от 0 до 9 с шагом 1. Таким образом мы переберём все возможные комбинации цифр, и будем проверять, делится ли число A нацело на $\overline{y2x}$. Для уменьшения количества операций число $\overline{y2x}$ необходимо вычислять внутри цикла по y . Что касается числа A , его следует вычислять пошагово, добавляя внутри каждого цикла слагаемое, зависящее от x, y и z соответственно.

алг Делимость

нач

цел x, y, z, num, n, den

$n = 20010002022004$

для x от 1 до 9 шаг 1

нц

$n1 = n + x * 10^{11} + x * 10^7 + x * 10^2$

для y от 0 до 9 шаг 1

нц

$n2 = n1 + y * 10^8 + y * 10$

$den = y * 10^2 + 20 + x$

для z от 0 до 9 шаг 1

нц

$num = n2 + z * 10^9$

если $num \bmod den = 0$ то

вывод x, y, z, num

всё

кц

кц

кц

кон

2. Разработайте алгоритм, который определяет (в порядке возрастания) номера разрядов, содержащих цифру 8 в десятичной записи числа 64^{256} . Нумерация разрядов начинается с нулевой степени десятичной записи. Например, для числа 1832_{10} первый подходящий разряд будет пронумерован номером 3.

Схема решения. Число 64^{256} слишком большое для того, чтобы его можно было точно представить в современном компьютере. Поэтому для решения задачи необходимо разработать способ хранения чисел нужной длины и алгоритм умножения чисел в новом представлении. Для представления чисел можно использовать массив, каждый элемент

Олимпиада школьников «Надежда энергетики». Заключительный этап. Очная форма.

которого хранит одну цифру числа. После нахождения значения 64^{256} можно будет просмотреть этот массив и найти номера разрядов, содержащих цифру 2. Для умножения чисел реализуем школьный алгоритм умножения «в столбик». Первое число надо умножить на каждую цифру второго числа, начиная с последней, и сложить результаты, учитывая, что результат умножения на очередную цифру надо сдвигать влево на соответствующее количество позиций. Для этого берётся первый сомножитель, помещается в промежуточную переменную, и каждая цифра первого сомножителя умножается на текущую цифру второго сомножителя. После умножения очередной цифры надо прибавить перенос из предыдущего разряда и вычислить собственно новую цифру как остаток от деления на 10 и число для переноса в следующий разряд как результат от деления на 10. Для ускорения работы алгоритма желательно отдельно рассмотреть случаи умножения числа на цифру 0 и цифру 1. После умножения выполняем сдвиг на нужное количество позиций и прибавляем результат в общую сумму. Для сложения чисел будем последовательно складывать цифры, начиная с последней. Если сумма двух цифр и переноса из предыдущего разряда больше или равна 10, то оставляем значение ($\langle \text{цифра1} \rangle + \langle \text{цифра2} \rangle + \langle \text{перенос} \rangle - 10$) и запоминаем наличие переноса в следующий разряд.

3. В волшебной стране Линии используется самобытная позиционная система счисления. В ней числа представляются следующим образом: цифра числа записывается в скобках (), а каждой цифре от 0_{10} до 9_{10} в привычной Вам десятичной с.с. соответствует число чёрточек «-», равное значению этой цифры. Разработайте алгоритм перевода натуральных чисел из шестнадцатеричной с.с. в с.с., принятую в Линии.

Решение. Пусть исходное число представлено в виде строки с шестнадцатеричными цифрами. Для формирования целочисленной переменной *num*, содержащей представленное значение, будем умножать текущее значение этой переменной на 16 и прибавлять значение очередной цифры. Начальное значение переменной *num* равно 0, цифры рассматриваются справа налево. Далее сформируем массив из цифр десятичного числа. Для этого будем последовательно запоминать остатки от деления переменной *num* на 10, а саму переменную делить на 10, пока она не станет равной 0. При выводе необходимо для каждой цифры вывести открывающую скобку, нужное количество чёрточек и закрывающую скобку.

алг Линии

нач

строка number

цел digits[0..1000], n, k, i, j

лог error

ввод number

// Перевод в число

n = 0

error = ложь

i = 1

пока i <= длина(number) и не error

нц

если '0' <= number[i] и number[i] <= '9' то

n = n * 16 + ord(number[i]) - ord('0')

иначе

если 'a' <= number[i] и number[i] <= 'f' то

n = n * 16 + ord(number[i]) - ord('a') + 10

иначе

если 'A' <= number[i] и number[i] <= 'F' то

n = n * 16 + ord(number[i]) - ord('A') + 10

иначе

error = истина

всё

всё

всё

i = i + 1

кц

если error то

вывод 'Ошибка в записи числа'

иначе

// Получение цифр

k = 0

```

повторять
  digits[k] = num mod 10
  k = k + 1
  num = num div 10
до num = 0

// Вывод цифр
для i от k - 1 до 0 шаг -1
нц
  вывод '('
  для j от 1 до digits[i]
  нц
    вывод '-'
  кц
  вывод ')'
кц
всё
кон

```

4. В археологических раскопках в Крыму при строительстве трассы «Таврида» археологи

нашли табличку с таким текстом: $\sqrt{2} = 1 + \frac{1}{2 + \frac{1}{2 + \frac{1}{2 + \dots}}}$

Пожалуйста, проверьте записанное – разработайте алгоритм вычисления $\sqrt{2}$ по схеме выше с точностью до 0.0001.

Решение. Если отбросить конец дроби (обозначенный многоточием), то последняя часть будет равна $1/2$. Положим переменную f_0 равной 0.5 и вычислим переменную f_1 по формуле $f_1 = 1 / (2 + f_0)$. Если разность по модулю между переменными f_0 и f_1 окажется меньше 0.0001, можно прекращать вычисления. Иначе положим переменную f_0 равной f_1 и снова вычислим f_1 по той же формуле. Итоговый результат равен $1 + f_1$.

Сравнивать необходимо именно два приближённых значения дроби, а не приближённое значение дроби со значением $\sqrt{2}$. Формула может оказаться неправильной, тогда разность между приближённым значением дроби и значением $\sqrt{2}$ никогда не станет меньше требуемой точности.

```

алг Дробь
нач
  вещ f0, f1

  f1 = 0
  повторять
    f0 = f1
    f1 = 1 / (2 + f0)
  до abs(f0 - f1) < 0.0001

  если abs((1 + f1) - sqrt(2)) < 0.0001 то
    вывод 'Формула верна'
  иначе
    вывод 'Формула не верна'
  всё
кон

```

5. Утверждения $A \rightarrow C$, $A \& B \rightarrow D$, $\neg B \rightarrow E$ истинны. Чему равны A и B , если C , D и E ложны?

Решение. Таблица истинности для логической функции «импликация» представлена ниже.

X	Y	$X \rightarrow Y$
ложь	ложь	Истина
ложь	истина	Истина
истина	ложь	ложь
истина	истина	истина

Из таблицы видно, что если следствие ложно, то для того, чтобы вся формула была истинной, необходимо, чтобы посылка также была ложна. Таким образом, утверждение A должно быть

Олимпиада школьников «Надежда энергетики». Заключительный этап. Очная форма.

ложно, в этом случае $A \& B$ также будет ложно. Кроме того, ложно должно быть $\neg B$, т.е. утверждение B должно быть истинно.