

**Материалы заданий Олимпиады школьников
«Надежда энергетики» по информатике
за 2011/2012 учебный год**

Характер и уровень сложности олимпиадных задач направлены на достижение целей проведения олимпиады: выявить способных участников, твердо владеющих школьной программой и наиболее подготовленных к усвоению образовательных программ технических ВУЗов, обладающих логикой и творческим характером мышления, умеющих алгоритмически описать реальные ситуации из различных предметных областей и применить к ним наиболее подходящие методы информатики. Необходимы знания способов описания алгоритмов (язык блок-схем, псевдокод) и умение работать с базовыми конструкциями.

Задания олимпиады дифференцированы по сложности и требуют различных временных затрат на полное и безупречное решение. Они охватывают все разделы школьной программы, но носят, в большинстве, комплексный характер, позволяющий варьировать оценки в зависимости от проявленных в решении творческих подходов и продемонстрированных технических навыков. Участники должны самостоятельно определить разделы и теоретические факты программы, применимые в каждой задаче, разбить задачу на подзадачи, грамотно выполнить решение каждой подзадачи, синтезировать решение всей задачи из решений отдельных подзадач.

Успешное выполнение олимпиадной работы не требует знаний, выходящих за пределы школьной программы, но, как видно из результатов олимпиады, доступно не каждому школьнику, поскольку требует творческого подхода, логического мышления, умения увидеть и составить правильный и оптимальный план решения, четкого и технически грамотного выполнения каждой части решения.

Умение справляться с заданиями олимпиады по информатике приходит к участникам с опытом, который вырабатывается на тренировочном и отборочном этапах Олимпиады.

Задание по информатике (10-11 класс) Вариант 2111

Задача 1

Согласно так называемому Закону Мура (предположение, высказанное в 1965 г. одним из основателей корпорации Intel Гордоном Муром), прогнозируемое количество транзисторов на чипе интегральных цифровых микросхем должно удваиваться каждые два года. Если предположить, что удвоение количества транзисторов в цифровых микросхемах оперативной памяти приводит к удвоению объема памяти, то к какому году объем оперативной памяти, выполненной в виде одной интегральной микросхемы должен достичь значения 1 гигабайт? В качестве отправной точки для прогноза принять выпущенную в 1968 году корпорацией Intel интегральную микросхему оперативной памяти с информационным объемом 1 килобит.

Задача 2

В приведенном примере на вычитание определить основание системы счисления q , цифры a, b , максимально возможные значения цифр c и d :

$$\begin{array}{r} _ a 2 4 d_q \\ - 1 b 7 c_q \\ \hline 1 5 5 0_q \end{array}$$

Задача 3

Студент европейского института человеческой истории собирается с помощью машины времени отправиться на практику в Англию конца 19 века. Он знает, что старая английская денежная система состояла из фунтов, шиллингов (12 шиллингов=1 фунт) и пенсов (20 пенсов = 1 шиллинг). Командировочные студента составляют 2 евро, 1 евро = 16 фунтов. Ваша задача - помочь узнать студенту, какими денежными средствами он обладает на практике в 19 веке.

Задача 4

Как известно из курса алгебры, даже уравнения полиномов степени выше 4 не имеют аналитического решения в общем виде. Тем более это относится к уравнениям общего вида. Для их приближенного решения используются численные методы, позволяющие найти решение x с заданной точностью ε . Один из них – метод Ньютона – состоит в следующем: для уравнения $f(x)=0$ приближение к решению на $(n+1)$ итерации ищется как $x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$. Значение x_0 -входной параметр. Останов происходит, когда $|x_{n+1} - x_n| < \varepsilon$. Пожалуйста, реализуйте алгоритм для решения уравнения $x^3 - \sin(x) = 4$.

Задача 5

Одна из первых ЭВМ в СССР “Сетунь” (50-е гг XX века), основывалась на троичной логике. В ней есть 3 логических значения: 0,1,2. Пожалуйста, опишите возможные недостатки и преимущества данной ЭВМ по сравнению с ЭВМ с двоичной логикой.

Задача 6

Девочки Маша и Саша придумали следующую игру: берется бумажное прямоугольное поле размером M на N клеток. Каждая из девочек по очереди загадывает случайным образом число от 1 до 10^3 и записывает его значение на клетку. Ее координаты получаются при попадании мячика, который кидает на поле вторая девочка. Если клетка уже занята, то ход пропускается. Маша записывает числа красной ручкой, а Саша – зеленой. Игра продолжается до тех пор, пока не заполнится все поле. Выигрывает девочка, у которой сумма записанных чисел больше. Перед Вами – завершенная партия. Кто выиграл? В качестве результата надо вывести имя девочки.

Задача 7

К девочкам Маше и Саше из задачи 6 после завершения игры пришел в гости мальчик Петя. Петя имеет хобби – упорядочение чисел (от меньшего к большему). Посмотрев на поле, Петя захотел как можно скорее упорядочить написанные на нем числа. Помогите Пете решить эту задачу.

Решения задач варианта

Задача 1

1 гигабайт = $1024 \cdot 1$ мегабайт = $1024 \cdot 1024 \cdot 1$ килобайт = $1024 \cdot 1024 \cdot 8 \cdot 1$ килобит = $= 2^{10} \cdot 2^{10} \cdot 2^3$ килобит = 2^{23} килобит.

Следовательно, количество транзисторов должно удвоиться 23 раза. Удвоение происходит раз в два года. Следовательно, потребуется 46 лет. $1968 + 46 = 2014$.

Задача 2

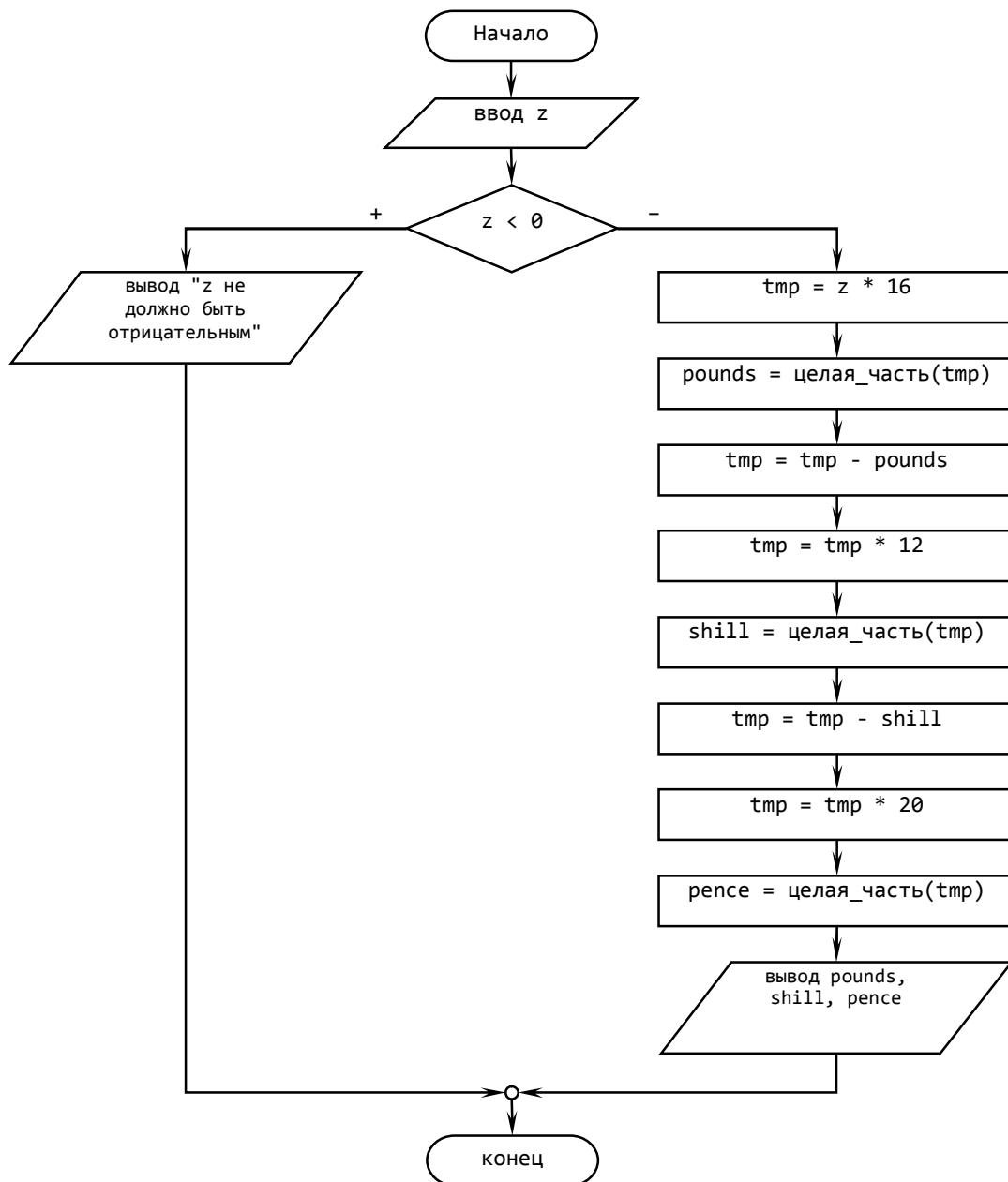
$$\begin{array}{r} _ a \ 2 \ 4 \ d_q \\ \underline{1 \ b \ 7 \ c_q} \\ 1 \ 5 \ 5 \ 0_q \end{array}$$

1. Первый разряд. $4 < 7$, поэтому нужен заём из второго разряда. $q + 4 - 7 = 5 \Rightarrow q = 8$.
2. Второй разряд. Из 2 был сделан заём, т.е. фактически остаётся 1. Из 1 вычитаем некоторое число b и получаем 5 – это возможно, только если $b > 1$ и будет сделан заём из третьего разряда. $2 - 1 + q - b = 5 \Rightarrow b = 2 - 1 + 8 - 5 = 4$.
3. Третий разряд. Из a был сделан заём, поэтому $a - 1 - 1 = 1 \Rightarrow a = 3$.
4. Нулевой разряд. $d - c = 0 \Rightarrow d = c$, и т.к. $q = 8$, то $d_{max} = c_{max} = 7$.

Задача 3

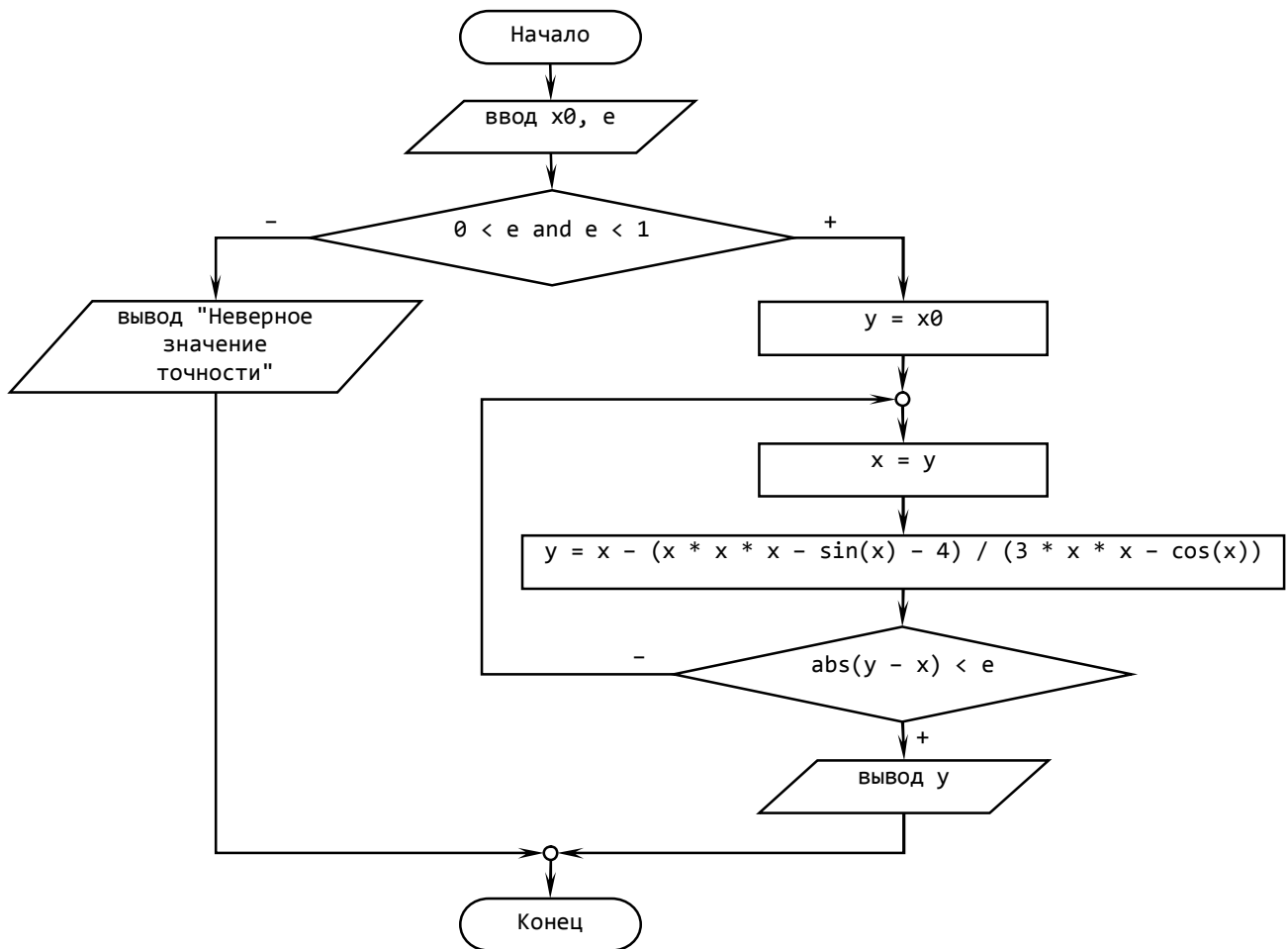
Прежде всего, проверим, что пользователь ввёл неотрицательное число.

1 евро = 16 фунтов, поэтому умножив z на 16, мы определим, сколько фунтов составляют z евро. Поскольку в общем случае z может быть вещественным, то и полученное после умножения число может быть не целым. Целая часть числа, полученного после умножения, будет представлять количество фунтов, полученных студентом, а дробную часть умножим на 12, поскольку 1 фунт = 12 шиллингов. После умножения снова возьмёт целую часть числа, которая будет представлять количество шиллингов, полученных студентом, а дробную часть умножим на 20, поскольку 1 шиллинг = 20 пенсов. После умножения снова возьмёт целую часть числа, которая будет представлять количество пенсов, полученных студентом, а дробную часть отбросим, т.к. её невозможно обменять даже на 1 пенс.



Задача 4

Вводим начальное приближение x_0 и точность ϵ . Переменная x будет хранить значение, полученное на n -ом шаге, а переменная y – значение, полученное на $(n + 1)$ -ом шаге. Для вычисления приближённого значения корня уравнения необходимо в цикле записать в переменную x значение корня, полученное на предыдущем шаге, т.е. значение переменной y , и затем вычислить новое приближение по формуле $y = x - f(x) / f'(x)$. Подставив в общую формулу конкретную функцию, для которой надо вычислить корень, и её производную, получим выражение $y = x - (x^3 - \sin(x) - 4) / (3x^2 - \cos(x))$. Вычисления прекращаем, когда $|y - x| < \epsilon$. Для того чтобы алгоритм работал корректно, перед началом цикла переменной y необходимо присвоить начальное приближение x_0 .



Задача 5

Преимущества: троичная система счисления имеет больший информационный объём, т.к. $\log_2 3 > \log_2 2$. Поэтому для хранения данных потребуется меньше места.

Недостатки: такую ЭВМ сложнее реализовать технически.

Задача 6

Числа, записанные девочками, будем хранить в матрице, причём для простоты хранения и обработки числа, записанные Машей, будем хранить без изменений, а числа, записанные Сашей, будем хранить, умножив на -1.

Вводим и проверяем размеры матрицы. Далее вводим числа и соответствующий цвет. Если цвет не красный и не зелёный, а также, если число не попадает в диапазон 1..1000, фиксируем наличие ошибки. Логическая переменная и сложные условия завершения циклов позволят нам прервать ввод сразу после нахождения ошибки. Если всё правильно, записываем в матрицу число без изменений или число, умноженное на -1. Таким образом, мы можем быть уверены, что матрица составлена правильно, и не осуществлять проверок в дальнейшем.

После ввода просматриваем матрицу и суммируем отдельно положительные и отрицательные числа. После просмотра всей матрицы сравниваем полученные суммы.

алг Игра (арг цел m, n; арг цел x[1..m, 1..n])

нач

цел i, j, num, sum1, sum2

лог error

сим color

```

ввод m, n
если m <= 0 или m > 10 или n <= 0 или n > 10 то
    вывод "Неверные размеры матрицы"
иначе
    error := false
    i := 1
    пока i <= m и не error
    нц
        j := 1
        пока j <= n и не error
        нц
            ввод num, color
            если num <= 0 или num > 1000 то
                error := true
            всё
            если color = 'к' то
                x[i, j] := num
            иначе
                если color = 'з' то
                    x[i, j] := -num
                иначе
                    error := true
            всё
        всё
        j := j + 1
    кц
    i := i + 1
кц
если error то
    вывод "Введены неверные данные"
иначе
    sum1 := 0
    sum2 := 0
    для i от 1 до m
    нц
        для j от 1 до n
        нц
            если x[i, j] > 0 то
                sum1 := sum1 + x[i, j]
            иначе
                sum2 := sum2 + x[i, j]
        всё
    кц
кц
sum2 := abs(sum2)
если sum1 = sum2 то
    вывод "Ничья"
иначе
    если sum1 > sum2 то
        вывод "Победила Маша"
    иначе
        вывод "Победила Саша"
    всё
всё
конец

```

Задача 7 (схема решения)

Одним из самых быстрых алгоритмов сортировки является алгоритм быстрой сортировки. Основная идея алгоритма состоит в том, что случайным образом выбирается некоторый элемент массива u , после чего массив просматривается слева, пока не встретится элемент $x[i]$ такой, что $x[i] \geq u$, а затем массив просматривается справа, пока не встретится

элемент $x[j]$ такой, что $x[j] \leq u$. Эти два элемента меняются местами, и процесс просмотра, сравнения и обмена продолжается, пока i не окажется больше j . В результате массив окажется разбитым на две части – левую, в которой значения элементов будут меньше или равны u , и правую, в которой значения элементов будут больше или равны u . Далее процесс рекурсивно продолжается для левой и правой частей массива до тех пор, пока каждая часть не будет содержать один элемент – массив из одного элемента по определению считается упорядоченным.

Поскольку алгоритм рассчитан на сортировку одномерного массива, матрицу придётся переписать в одномерный массив, а после сортировки – обратно.