

ЗАДАНИЕ ПО ИНФОРМАТИКЕ
ВАРИАНТ 32111 для 11-го класса

Разрабатывать алгоритмы необходимо на языке блок-схем, псевдокоде или естественном языке

1. Предложите алгоритм, как установить, что заданные три точки (координаты точек заданы парами чисел с плавающей точкой) гарантированно не лежат на одной и той же окружности.

Решение: достаточно установить, что три точки лежат на одной прямой, то есть не являются вершинами треугольника. Одним из способов является вычисление расстояний между парами точек по формуле $r_{ij} = \sqrt{(x[i]-x[j])^2 + (y[i]-y[j])^2}$ и проверка, выполняется ли неравенство треугольника – каждая из сторон меньше суммы двух других.

2. Микроконтроллер должен вычислить сумму произведений двадцати шести пар 14-разрядных чисел со знаком (1 двоичный разряд выделяется на знак). Возможно ли провести вычисления, используя 32-битные регистры для хранения операндов и результата?

Решение: произведение требует 26 разрядов + 1 знаковый разряд. Суммирование 26 пар может добавить еще 5 разрядов, таким образом, требуется как раз 32 разряда. Следовательно, ответ – да, возможно.

3. Необходимо оценить успеваемость учеников школы по триместровым оценкам. Предложите алгоритм расчета процента учеников, имеющих наилучшую динамику изменения среднего триместрового балла в процентном отношении, по результатам второго триместра.

Решение (вариант): Пусть N – число учеников, массив из 11 элементов $p[i]$ – число предметов (оценок) за год для 11 классов. В массиве $A[i,j,k]$ хранятся оценки учеников (размер массива $N \times n \times 2$, где $n = \max(p[i])$, при этом предполагаем, что массив A предзаполнен нулями). Достаточно рассчитать средний балл каждого ученика за триместр, вычислить разность между первым и вторым триместрами и затем отсортировать массив приращений баллов по возрастанию (убыванию), тогда можно выбрать всех учеников с произвольно заданным приращением триместрового балла (ball).

Ввод $N, n, A, ball$

Average=массив[$N \times 2$]

Для $p=1$ до 2

 Для $i = 1$ до N

$k=0$

$S=0$

 Для $j = 1$ до n

$S=S+A[i,j,p]$

 Если $A[i,j,p] \neq 0$ То $k=k+1$

 Конец

 Average[i,p]= S/k

 Конец

Конец

Delta=массив[N]

Для $i = 1$ до N

 Delta[i]= $100 \cdot (Average[i,2] - Average[i,1]) / Average[i,1]$

Конец

$S=0$

Для $j=1$ До $N-1$

 Для $i=0$ До $N-1-j$

 Если Delta[i] > Delta[$i+1$] То

Олимпиада школьников «Надежда энергетики». Отборочный этап. Заочная форма.

```
S=Delta[i]
Delta[i]= Delta[i+1]
Delta[i+1]=S
Конец
Конец
Конец
Для i = 1 до Целая часть(N/2)
    Если Average[i]>M To
        S=S+Average[i]
        k=k+1
Конец
u=0
Для i = 1 до N
    Если Delta[i]>ball To
        u=u+1
Конец
Вывод (100*u/N)
```

4. При анализе устойчивости технических систем возникает задача сравнения максимумов и минимумов почти периодических колебаний. Предложите алгоритм расчета отношения соседних максимума и минимума колебаний, если исследуемый сигнал задан в виде массива из N значений с плавающей точкой, так, что в пределах массива гарантированно поместятся не менее двух периодов сигнала.

Решение (вариант):

Пусть $X[i]$ – массив из N элементов, чисел с плавающей точкой. Обратим внимание на то, что мы не можем установить, является ли экстремумом (максимумом или минимумом) первая или последняя точка массива.

Max=0

Min=0

k=0

m=0

Для i=2 До N-1

Если $(X[i]>X[i-1])$ И $(X[i]>X[i+1])$ To

Max=X[i]

k=k+1

Конец

Если $(X[i]<X[i-1])$ И $(X[i]<X[i+1])$ To

Min=X[i]

m=m+1

Конец

Если $(k>0)$ И $(m>0)$ To

Если $Min \neq 0$ To Вывод(Max/Min) Иначе Вывод(«Деление на 0»)

k=0

m=0

Конец

Конец

5. Предложите алгоритм генерации случайной таблицы из 4×5 символов (всего 16 вариантов символов с кодами 0-15), такой, чтобы гарантировать отсутствие повторения символов в строке и столбце. При разработке считайте, что необходимо минимально возможное число раз использовать генератор случайных чисел (выдает случайное число с плавающей точкой в диапазоне 0 - 1).

Решение (вариант):

поскольку условием является минимизация числа вызовов генератора случайных чисел, целесообразно при расчете каждого нового элемента таблицы исключать из списка доступных символов те, которые уже присутствуют в строке и в столбце. Пусть $A[i,j]$ – искомая таблица кодов символов, будем заполнять ее по столбцам. В массиве $V[i]$ будем хранить список символов. Потребуется вспомогательный массив $C[i]$, содержащий список неиспользованных символов.

Для $i=1$ До 4

 Для $j=1$ До 5

$C[] = V[]$

$S=0$

$n=16$

 Для $ii=1$ До $i-1$

$S=C[n]$

$C[n]=C[A[ii,j]]$

$C[A[ii,j]]=S$

$n=n-1$

 Конец

 Для $jj=1$ До $j-1$

$S=C[n]$

$C[n]=C[A[i,jj]]$

$C[A[i,jj]]=S$

$n=n-1$

 Конец

$A[i,j] = \text{Целая часть}(\text{random}()*(n-1))+1$

 Конец

Конец