

ЗАДАНИЕ ПО ИНФОРМАТИКЕ
ВАРИАНТ 31101 для 10-го класса

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

1. Предложите алгоритм, как установить, что заданные четыре точки (координаты точек заданы парами чисел с плавающей точкой) являются вершинами квадрата.

Решение (вариант):

Одним из способов является вычисление расстояний между парами точек по формуле $r_{ij} = \sqrt{(x[i]-x[j])^2 + (y[i]-y[j])^2}$, всего 6 сочетаний, и проверка, выполняется ли в совокупности следующий набор условий – четыре расстояния оказываются равными друг-другу, еще два – также равными друг-другу, но при этом – строго больше предыдущих четырех (последняя проверка позволяет исключить вырожденный случай, когда все шесть расстояний равны нулю).

2. Микроконтроллер должен вычислить сумму произведений девятнадцати пар 13-разрядных чисел. Возможно ли провести вычисления, используя 32-битные регистры для хранения операндов и результата?

Решение: произведение $2^{13} * 2^{13}$ требует $2^{(13+13)} = 2^{26}$ – 26 разрядов. Суммирование 19 пар может добавить еще 5 разрядов, таким образом, требуется 31 разряд, что меньше, чем 32. Следовательно, ответ – да, возможно.

3. Необходимо оценить успеваемость учеников школы по годовым оценкам. Предложите алгоритм расчета медианного балла (такого, чтобы половина учеников имела балл выше медианного).

Решение (вариант): Пусть N – число учеников, массив из 11 элементов $p[i]$ – число предметов (оценок) за год для 11 классов. В массиве $A[i,j]$ хранятся оценки учеников (размер массива $N \times n$, где $n = \max(p[i])$, при этом предполагаем, что массив A предзаполнен нулями). Достаточно рассчитать средний балл каждого ученика и затем отсортировать массив средних баллов по возрастанию (убыванию). В середине отсортированного массива мы обнаружим медианный балл.

Ввод N,n,A

Average=массив[N]

Для i = 1 до N

 k=0

 S=0

 Для j = 1 до n

 S=S+A[i,j]

 Если A[i,j]! = 0 То k=k+1

 Конец

 Average[i]=S/k

Конец

S=0

Для j=1 До N-1

 Для i=0 До N-1-j

 Если Average[i] > Average[i+1] То

 S=Average[i]

 Average[i]= Average[i+1]

 Average[i+1]=S

 Конец

 Конец

Конец

$u=0$

Если $\text{Остаток}(N/2) \neq 0$ То $u=0.5 * (\text{Average}[\text{Целая часть}(N/2)+1] + \text{Average}[\text{Целая часть}(N/2)])$

Иначе $u = \text{Average}[\text{Целая часть}(N/2)+1]$

Вывод (u)

4. При анализе трафика и оптимизации маршрутизации в сетях необходимо убедиться в отсутствии циклических маршрутов в сети. Предложите алгоритм поиска циклических маршрутов, если каждое устройство имеет список устройств, соединенных непосредственно с ним, для опроса устройства необходимо иметь доступ к нему (построить к нему маршрут от исходного устройства).

Решение (вариант):

Необходимо в цикле пройти по устройствам, составляя общий список пройденных устройств. Добавляя каждый новый элемент списка следует проверять, присутствует ли данное устройство в списке. В случае, если возникает дублирование устройства в списке пройденных, фиксируется факт наличия замкнутого контура.

5. Предложите алгоритм генерации случайной таблицы из 6×6 символов (256 вариантов символов с кодами 0-255) такой, чтобы гарантировать отсутствие повторения символов. 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 необходимы 36 случайных символов, минимальное число вызовов генератора случайных чисел равно 36. Для исключения повторов следует при выборе очередного символа исключать его из выбора в дальнейшем:

Пусть коды символов находятся в массиве $A[i]$

$V = \text{массив}[6,6]$

$n=256$

Для $i=1$ До 36

$k = \text{Целая часть}(\text{random}() * (n-1)) + 1$

$V[\text{Целая часть}((i-1)/6)+1, \text{Остаток}((i-1)/6)+1] = A[k]$

$S = A[k]$

$A[k] = A[n]$

$A[n] = S$

$n = n - 1$

Конец