ЗАДАНИЕ ПО ИНФОРМАТИКЕ

ВАРИАНТ 37111 для 11-го класса

1. Кластеризация данных – можно ли разделить массив n точек в пространстве на m частей так, чтобы расстояние всех элементов от среднего в пределах части (кластера) было меньше половины расстояния между средними значениями любой пары частей? Предложите алгоритм проверки для заданного массива точек, m и n.

__Ответ:__ возможны варианты решения. Целесообразным представляется выделить функцию, возвращающую расстояние между точками: R[i,j]=sqrt((x[i]-x[j])^2+ (y[i]-y[j])^2+ (z[i]-z[j])^2) и сформировать группы (кластеры) точек, по p[i] = Целая часть (n / m) элементов в каждом, добавив оставшиеся точки в последний кластер, число элементов в нем p[m]=p[m-1] + Остаток (n/m). Первоначальное распределение точек по кластерам может быть произвольным.

Средняя точка в пределах кластера может быть вычислена как:

k=1

Цикл по i=1 до m (по кластерам)

M[i]=0

Цикл по j=1 до p[i] (по элементам кластера)

XM [i]=XM[i]+x[k]

YM [i]=YM[i]+y[k]

ZM [i]=ZM[i]+z[k]

k=k+1

Конец j=j+1

XM[i]=XM[i]/p[i] (среднее значение для кластера)

YM[i]=YM[i]/p[i] (среднее значение для кластера)

ZM[i]=ZM[i]/p[i] (среднее значение для кластера)

Конец i=i+1

Проверка условия при этом выглядит так:

k=1

Condition = Истина

Cmin= sqrt((XM[1]-XM[m])^2+ (XM[1]-YM[n])^2+ (ZM[1]-ZM[n])^2)

Цикл по i=1 до m (по кластерам)

Цикл по j=1 до m (по кластерам)

C=Cmin

Если i~=j то C= sqrt((XM[i]-XM[j])^2+ (YM[i]-YM[j])^2+ (ZM[1]-ZM[n])^2)

Если C<Cmin то Cmin=C

Цикл по i=1 до m (по кластерам)

Dmax=0

Цикл по j=1 до p[i] (по элементам кластера)

D= sqrt((XM[i]-x[k])^2+ (YM[i]-y[k])^2+ (ZM[1]-z[k])^2)

Если D>Dmax то Dmax=D

k=k+1

Конец j=j+1

Если Cmin<2*Dmax то Condition = Ложь (проверка условия для каждого кластера)

Конец i=i+1

Если проверка дала неудовлетворительный результат можно перераспределить элементы между кластерами и повторять проверку условия и перераспределение элементов до тех пор, пока не образуются вырожденные (пустые кластеры). При этом условие исключения точки из кластера – максимальное расстояние от точки до центра кластера, условие добавление в кластер – минимизация расстояния до центра кластера.

2. Классификация по методу опорных векторов – построить алгоритм проверки принадлежности заданной точки (x,y,z,s) в гиперпространстве классу – нижнему или верхнему гипер-полупространству относительно гиперплоскости-классификатора. Обученный классификатор e = ax+by + cz+ds

<u>Ответ:</u> достаточно проверить условие

Low = Истина (принадлежность ниже границы)

Если s > (e-a*x-b*y-c*z)/d то Low = Ложь

High = Истина (принадлежность нижней полуплоскости)

Если s < (e-a*x-b*y-c*z)/d то High = Ложь

Соответственно, при строгом выполнении равенства, нельзя вынести определенного суждения о принадлежности точки определенному классу

3. Предложите алгоритм суммирования двух трехразрядных чисел с использованием только логических функций ИЛИ, НЕ

<u>Ответ:</u>

Пусть A0, A1,A2 –разряды первого операнда, B0, B1, B2 –разряды второго операнда, S0, S1, S2, S4 – разряды суммы

Тогда S0 = (A0 *И* (НЕ B0)) ИЛИ (B0 *И* (НЕ A0)) = (НЕ ((A0 ИЛИ (НЕ B0)) ИЛИ НЕ(B0 ИЛИ (НЕ A0)))

перенос C = A0 И B0 = НЕ (НЕ(A0) ИЛИ НЕ (B0))

S1 = (A1 И (НЕ (B1) И (НЕ (C)) *ИЛИ* (B1 И (НЕ (A1) И (НЕ (C)) *ИЛИ* (C И (НЕ (B1) И (НЕ (A1)) *ИЛИ* (A1 И B1 И C)

C1 = (A1 И B1) ИЛИ (A1 И C) ИЛИ (B1 И C)

S2 = (A2 И (НЕ (B2) И (НЕ (C1)) *ИЛИ* (B2 И (НЕ (A2) И (НЕ (C1)) *ИЛИ* (C1 И (НЕ (B2) И (НЕ (A2)) *ИЛИ* (A2 И B2 И C1)

S3 = (A2 И B2) ИЛИ (A2 И C1) ИЛИ (B2 И C1)

причем каждая операция А *И* В заменяется на НЕ (НЕ(А) ИЛИ НЕ(В))

4. Для суммирования модулированной последовательности чисел без потери точности предлагается хранить результат в 256-разрядном числе S. Какова длина последовательности 8-разрядных чисел, такая, что сумма сможет быть размещена в S, если модуляция последовательности осуществляется перемножением элемента на случайно выбранное число из массива [0, -1, 2, -3, 8, 16].

Ответ: предполагая двоичные разряды, получаем наибольшее число со знаком, равное 2^{255}-1. В худшем случае каждое число в последовательности имеет 7 разрядов, равных 1, то есть 2^7-1, умножается на 16 (т.е. сдвигается на четыре двоичных разряда). Число элементов последовательности будет равно отношению 2^{251}-1 к 2^7-1.

5. На балансе предприятия ТЭЦ имеется сеть теплоснабжения, состоящая из m участков труб и сеть горячего водоснабжения (ГВС), состоящая из n участков труб. Все трубы одинаково изношены.

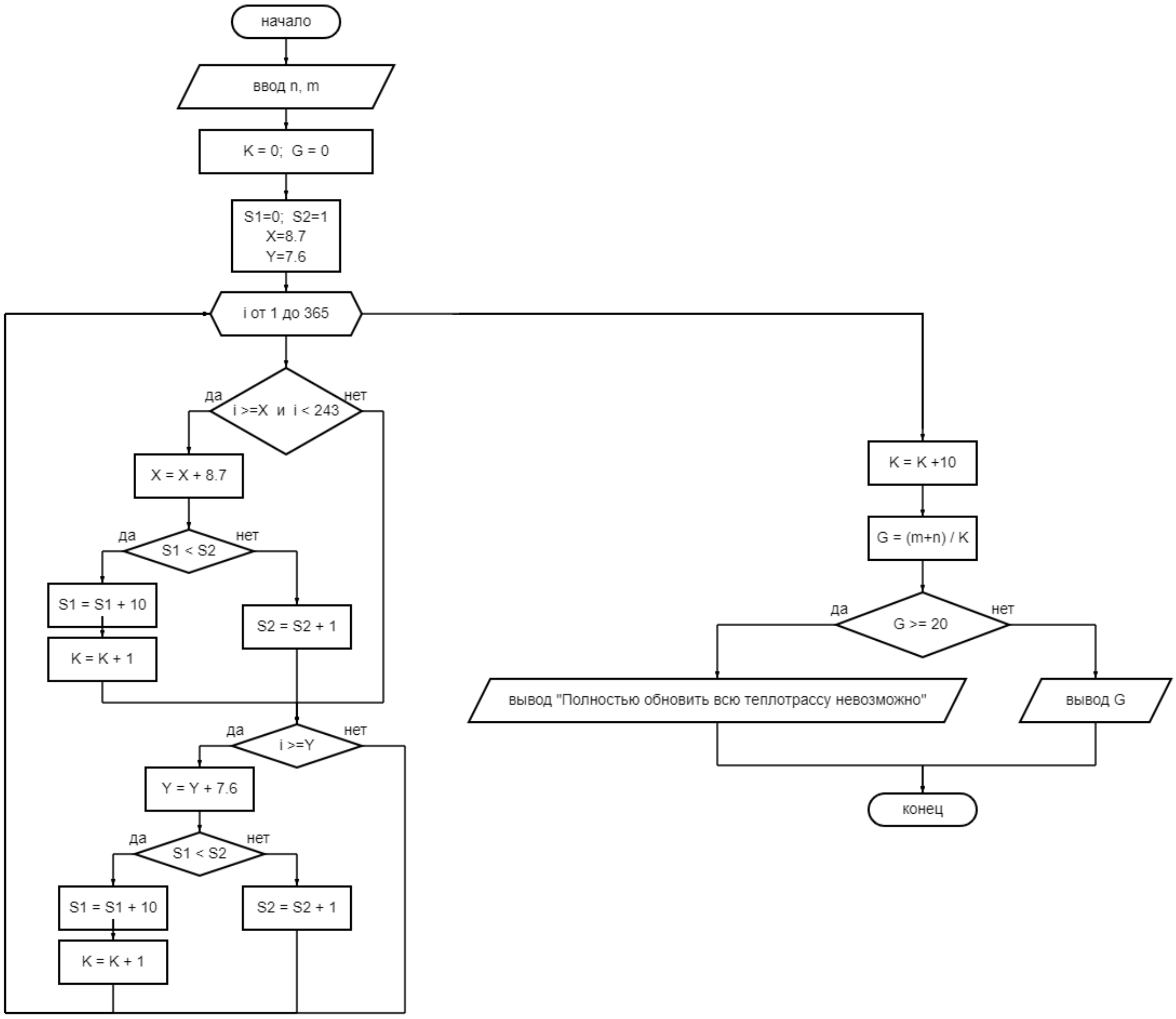
Нормальное время эксплуатации металлических труб около 20 лет. Однако замена всех сетей не рентабельна для предприятия. По причине естественного износа на трубах периодически возникают прорывы.

На предприятии для ремонта сетей имеется две бригады: одна бригада полностью меняет участок трубы на новый, затрачивая на ремонт десять дней, а другая за один день просто ставит временную латку. На ремонт поврежденного участка выезжает та бригада, которая освободилась ранее.

В период эксплуатации аварии на сети теплоснабжения возникают 7 раз в 2 месяца, а на сети ГВС 4 раза в месяц.

Когда, наконец, будет обновлена вся сеть теплоснабжения и ГВС, если учесть, что трубы ГВС эксплуатируются круглый год, а сеть теплоснабжения только в отопительный период (с сентября по апрель), зато в летний период (4 месяца) производится полноценный ремонт 10 участков теплосети, выполняемый дополнительной бригадой.

Представьте в виде блок-схемы алгоритм работы программы, определяющей через сколько месяцев будет произведена полная замена трубопроводов, если положить, что начался первый месяц отопительного периода и на первую аварию выезжает первая бригада.


начало
ввод n, m
K = 0; G = 0
S1=0; S2=1
X=8.7
Y=7.6
i от 1 до 365
да
i >=X и i < 243
нет
X = X + 8.7
да
S1 < S2
нет
S1 = S1 + 10
K = K + 1
S2 = S2 + 1
да
i >=Y
нет
Y = Y + 7.6
да
S1 < S2
нет
S1 = S1 + 10
K = K + 1
S2 = S2 + 1
K = K +10
G = (m+n) / K
да
G >= 20
нет
вывод "Полностью обновить всю теплотрассу невозможно"
вывод G
конец