

Цифровые сенсорные системы

2022/23 учебный год

Второй отборочный этап

Задача IV.1. Эмуляция работы протокола IIC (50 баллов)

Темы: передача данных, датчики, массивы.

Условие

Микроконтроллер и датчик соединены между собой информационном каналом посредством протокола IIC. Необходимо произвести дешифрацию последовательности.

Формат входных данных

На вход разрабатываемой функции приходит последовательность битов, зафиксированных на линии SDA во время передачи данных от датчика (в режиме ведомого) к контроллеру (в режиме ведущего). Известно, что метеодатчик отправляет данные о температуре, влажности и давлении в виде чисел в беззнаковом целом представлении, объемом 2 байта каждое. Также, в последовательности содержится номер датчика.

Формат выходных данных

На выходе ожидается номер датчика и 3 числа, характеризующие температуру, влажность и давление, разделенных пробелом.

Примеры

Пример №1

Стандартный ввод
0100101000110101110101010000010101010011010100001101000110001000
Стандартный вывод
[74, 55208, 21866, 13508]

Пример программы-решения

Ниже представлено решение на языке Python 3.

```
1 from random import randint
2
3 def SDA_reader(inStr):
4     id_SDA = inStr[1:8]
5     bytesStr = []
6     position = 10
7     for pos in range(6):
```

```

8         bytesStr.append(inStr[position:position+8])
9         position = position + 9
10        print(id_SDA, bytesStr)
11        id_SDA = int(id_SDA, 2)
12        x = str(bytesStr[0]) + str(bytesStr[1])
13        y = str(bytesStr[2]) + str(bytesStr[3])
14        z = str(bytesStr[4]) + str(bytesStr[5])
15
16        out = []
17        out.append(id_SDA)
18        out.append(int(x, 2))
19        out.append(int(y, 2))
20        out.append(int(z, 2))
21
22        return out
23
24    def bitFiller(number):
25        out = str(bin(number))
26        out = out[2:len(out)]
27        while len(out) % 16 != 0:
28            out = '0' + out
29
30        bytesStr = []
31        bytesStr.append(out[0:8])
32        bytesStr.append(out[8:16])
33
34        return bytesStr
35
36    def SDA_compiler(idNum, numsToSend):
37        idStr = bitFiller(idNum)
38        idStr = idStr[1]
39        idStr = idStr[1:len(idStr)]
40        bytesOfNumbers = []
41        for i in range(len(numsToSend)):
42            bytesOfNumbers.append(bitFiller(numsToSend[i]))
43        out = '0' + idStr + "00"
44        for i in range(3):
45            for j in range(2):
46                out = out + str(bytesOfNumbers[i][j]) + "0"
47        return out
48
49    idNum = randint(0, 127)
50    print("ID = ", idNum)
51    numsToSend = []
52    for i in range(3):
53        numsToSend.append(randint(0, 65536) )
54        print("Num", i, "=", numsToSend[-1])
55    print(SDA_compiler(idNum, numsToSend))
56
57    print(SDA_reader(SDA_compiler(idNum, numsToSend)))

```

Тестовые данные

Тестовые данные доступны по ссылке: <https://disk.yandex.ru/d/7PE693DjbLd rVw>.

Задача IV.2. Конечные автоматы на практике (50 баллов)

Темы: программирование, алгоритмы, массивы.

Условие

Предприятие по производству полимеров выполняет заказ на приготовление раствора-праймера. Микроконтроллер осуществляет мониторинг температуры приготавливаемого раствора и управление приводом перемешивания раствора и нагревателем ёмкости. Показания температуры снимаются с трех уровней высоты емкости: нижний, серединный и верхний. Перемешивание раствора необходимо для предотвращения перегрева части раствора, а также во избежание застывания. Нагреватель требуется для подогрева раствора.

Задача: в зависимости от входных данных необходимо управлять нагревателем и приводом перемешивания раствора таким образом, чтобы приблизить состояние раствора к уставке (нагреть или остудить). Если раствор нагрет неоднородно необходимо перемешать его — в таком случае, в отсутствие дополнительного нагрева, температуру считать по всему объему как среднее арифметическое между 3 величинами, полученными с датчиков.

Формат входных данных

Входные данные представляют собой 4 числа: уставку температуры и 3 показания датчиков температуры в градусах Цельсия.

Формат выходных данных

Выходные данные: представляют собой список состояний для нагревателя и привода перемешивания.

Включенное состояние обозначить единицей, выключенное — нулем.

Порядок заполнения состояний следующее: нагреватель, привод перемешивания.

Примеры

Пример №1

Стандартный ввод
200 205 200 190
Стандартный вывод
[1, 1]

Решение

Нахождение средней уставки (среднего арифметического) температур и сравнение его с заданной уставкой. Вывод необходимого массива согласно условию со значениями 1 и 0.

Пример программы-решения

Ниже представлено решение на языке Python 3.

```
1 import numpy as np
2 int_list = []
3 for i in input().split():
4     int_list.append(int(i))
5
6 temp = int_list.pop(0)
7 mean = int(np.mean(int_list))
8
9 def sum(mean, temp):
10     ans = []
11     if mean == temp:
12         ans = [1,0]
13     elif mean < temp:
14         ans = [1,1]
15     elif mean > temp:
16         ans = [0, 1]
17     return ans
18
19 print(sum(mean, temp))
```

Тестовые данные

Пример №1

Стандартный ввод
200 205 200 190
Стандартный вывод
[1, 1]

Пример №2

Стандартный ввод
230 220 220 220
Стандартный вывод
[1, 0]

Пример №3

Стандартный ввод
80 14 0 -13
Стандартный вывод
[1, 1]

Пример №4

Стандартный ввод
130 200 190 -150
Стандартный вывод
[1, 1]

Пример №5

Стандартный ввод
197 179 202 217
Стандартный вывод
[0, 1]

Пример №6

Стандартный ввод
199 60 169 25
Стандартный вывод
[1, 1]

Пример №7

Стандартный ввод
82 126 240 15
Стандартный вывод
[0, 1]

Пример №8

Стандартный ввод
40 60 58 62
Стандартный вывод
[0, 1]

Пример №9

Стандартный ввод
143 149 202 27
Стандартный вывод
[1, 1]

Пример №10

Стандартный ввод
0 0 0 0
Стандартный вывод
[0, 0]