

Christy is interning at HackerRank. One day she has to distribute some chocolates to her colleagues. She is biased towards her friends and plans to give them more than the others. One of the program managers hears of this and tells her to make sure everyone gets the same number.

To make things difficult, she must equalize the number of chocolates in a series of operations. For each operation, she can give **1, 2 or 5** pieces to all but one colleague. Everyone who gets a piece in a round receives the same number of pieces.

Given a starting distribution, calculate the minimum number of operations needed so that every colleague has the same number of pieces.

Example

arr = [1, 1, 5]

arr represents the starting numbers of pieces for each colleague. She can give **2** pieces to the first two and the distribution is then [3, 3, 5]. On the next round, she gives the same two **2** pieces each, and everyone has the same number: [5, 5, 5]. Return the number of rounds, **2**.

Function Description

Complete the *equal* function in the editor below.

equal has the following parameter(s):

- *int arr[n]*: the integers to equalize

Returns

- *int*: the minimum number of operations required

Input Format

The first line contains an integer *t*, the number of test cases.

Each test case has **2** lines.

- The first line contains an integer *n*, the number of colleagues and the size of *arr*.
- The second line contains *n* space-separated integers, *arr[i]*, the numbers of pieces of chocolate each colleague has at the start.

Constraints

$1 \leq t \leq 100$
 $1 \leq n \leq 10000$

The number of chocolates each colleague has initially < **1000**.

Sample Input

STDIN	Function
-----	-----

```
1         t = 1
4         arr[] size n = 4
2 2 3 7    arr =[2, 2, 3, 7]
```

Sample Output

2

Explanation

Start with **[2, 2, 3, 7]**

Add **1** to all but the 3rd element → **[3, 3, 3, 8]**

Add **5** to all but the 4th element → **[8, 8, 8, 8]**

Two operations were required.

Sample Input 1

```
1
3
10 7 12
```

Sample Output 1

3

Explanation 1

Start with **[10, 7, 12]**

Add **5** to the first two elements → **[15, 12, 12]**

Add **2** to the last two elements → **[15, 14, 14]**

Add **1** to the last two elements → **[15, 15, 15]**

Three operations were required.