

PXQRTZ

Jogo

Você e seu amigo estão se divertindo, jogando um jogo de tabuleiro. Nele, cada jogador tem um baralho de cartas e ambos têm a mesma quantidade de cartas. Estas cartas representam personagens que se enfrentam em diversas lutas. Então ocorrerá sucessivas lutas, onde um personagem de um dado jogador vai enfrentar um personagem do oponente. Como regra, cada personagem irá participar de uma única luta. Ao final do jogo, ganha o jogador que mais vezes derrotar seu oponente. Cada baralho é formado por diversos personagens, e cada um tem certa pontuação de luta. Em cada luta de personagens, vence aquele com maior pontuação de luta. Se ambos têm a mesma pontuação de luta, declara-se empate e ninguém vence. Sua tarefa é: conhecendo a pontuação de luta das suas cartas e das cartas do seu oponente, calcule o número máximo de lutas que você pode ganhar se os seus personagens forem confrontados em uma ordem apropriada.

Input

Em cada entrada há vários casos de teste. Cada um seguindo a forma: primeira linha contém um número que representa a quantidade de cartas de cada jogador ($1 \leq C \leq 10^5$). A segunda linha contém C números, que representam as pontuações de luta dos personagens do seu oponente ($1 \leq C_i \leq 10^9$). A terceira linha contém outros C números, que representam as pontuações de luta dos seus próprios personagens.

Output

Para cada caso de teste, imprima uma linha contendo o número **máximo** de lutas que você pode vencer.

Restrições da Solução C

- Seu código não pode usar rotinas de biblioteca para manipular/alterar os dados.
 - Apenas as rotinas de biblioteca p/ interação com o teclado/tela (ex., printf, scanf, gets, ...) e as rotinas relacionadas à alocação de memória (ex. malloc, calloc, ...) são permitidas.
- Seu código não deve usar variáveis globais.
- Seu programa deve usar alocação dinâmica de memória
- Seu código deve ser formado por funções e/ou procedimentos criados por você.
 - Deve haver no mínimo 1 funções/procedimentos.

Example

Input:

```
7
3 87 59 42 11 17 68
20 55 9 71 4 7 81
```

8
99 85 95 38 42 20 20 92
73 33 52 24 83 49 40 38

Output:

5
4

Example

Input:

8
63 10 28 3 17 77 99 41
83 28 73 35 18 88 19 57
9
86 27 43 9 87 89 51 17 72
28 42 21 69 83 49 32 83 12

Output:

7
6