

Torre de Dados

Nome do arquivo: “`torre.x`”, onde `x` deve ser `c`, `cpp`, `pas`, `java`, `js`, `py2.py` ou `py3.py`

Hortência está brincando de construir uma torre com dados de seis faces. Os dados são similares aos dados comuns utilizados em jogos, com as faces estampadas com valores de 1 a 6. Mas os dados usados por Hortência têm uma grande diferença em relação aos dados comuns: enquanto em dados comuns a soma dos valores em faces opostas é sempre sete, para os dados de Hortência os valores em faces opostas nem sempre têm soma sete.

Hortência está criando a torre empilhando os seus dados, obedecendo às seguintes regras:

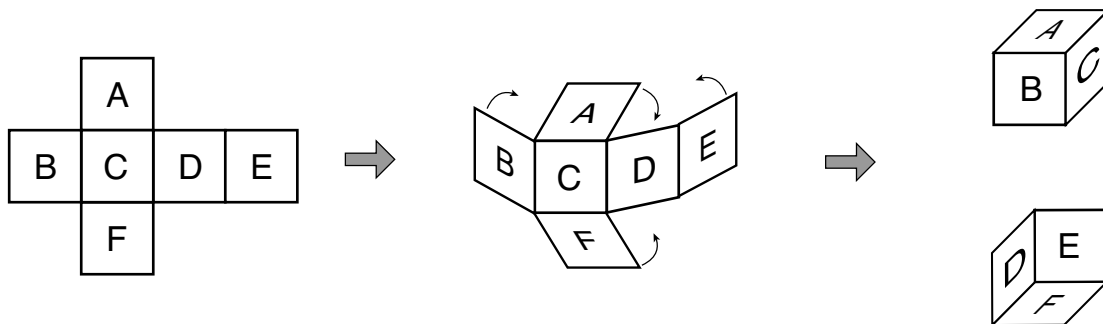
1. Seja X dado que está colocado imediatamente em cima de um dado Y . Então o valor da face inferior de X deve ser igual ao valor da face superior de Y . Por exemplo, se a pilha tem três dados R , S e T , empilhados nessa ordem de baixo para cima, então o valor da face superior de R deve ser igual ao valor da face inferior de S , e o valor da face superior de S deve ser igual ao valor da face inferior de T .
2. Os dados devem ter suas laterais alinhadas, ou seja, a torre formada tem exatamente quatro lados, correspondendo aos lados dos dados.

Hortência quer criar uma torre tal que a soma dos valores de um dos lados da torre seja a maior possível. Note que após empilhar os dados obedecendo à regra (1), Hortência pode girar cada dado horizontalmente de forma independente, obedecendo à regra (2), para alterar os valores das somas dos lados.

Dadas as informações sobre os dados utilizados na torre, escreva um programa para determinar o maior valor possível para a soma de um dos lados da torre.

Entrada

A primeira linha da entrada contém um inteiro N , o número de dados. Cada uma das N linhas seguintes descreve um dado e contém seis inteiros A , B , C , D , E e F , identificando os valores dos lados do dado, conforme a figura abaixo. *Errata, faltou a seguinte frase na prova online: os dados são empilhados na torre na ordem dada na entrada, de baixo para cima. As submissões serão julgadas com dois conjuntos de testes, um considerando os dados na ordem dada na entrada e outro considerando que a ordem pode ser qualquer uma e o competidor receberá a maior pontuação entre as duas correções.*



Saída

Seu programa deve produzir uma única linha, contendo um único inteiro, o maior valor possível para a soma de um dos lados da torre.

Restrições

- $1 \leq N \leq 1000$
- $1 \leq A, B, C, D, E, F \leq 6$, todos distintos em um dado.

Exemplos

Exemplo de entrada 1 2 1 2 3 4 5 6 1 2 6 4 5 3	Exemplo de saída 1 12
Exemplo de entrada 2 5 4 6 1 5 3 2 1 3 2 4 6 5 5 3 1 4 2 6 6 1 5 2 4 3 4 6 1 5 3 2	Exemplo de saída 2 28