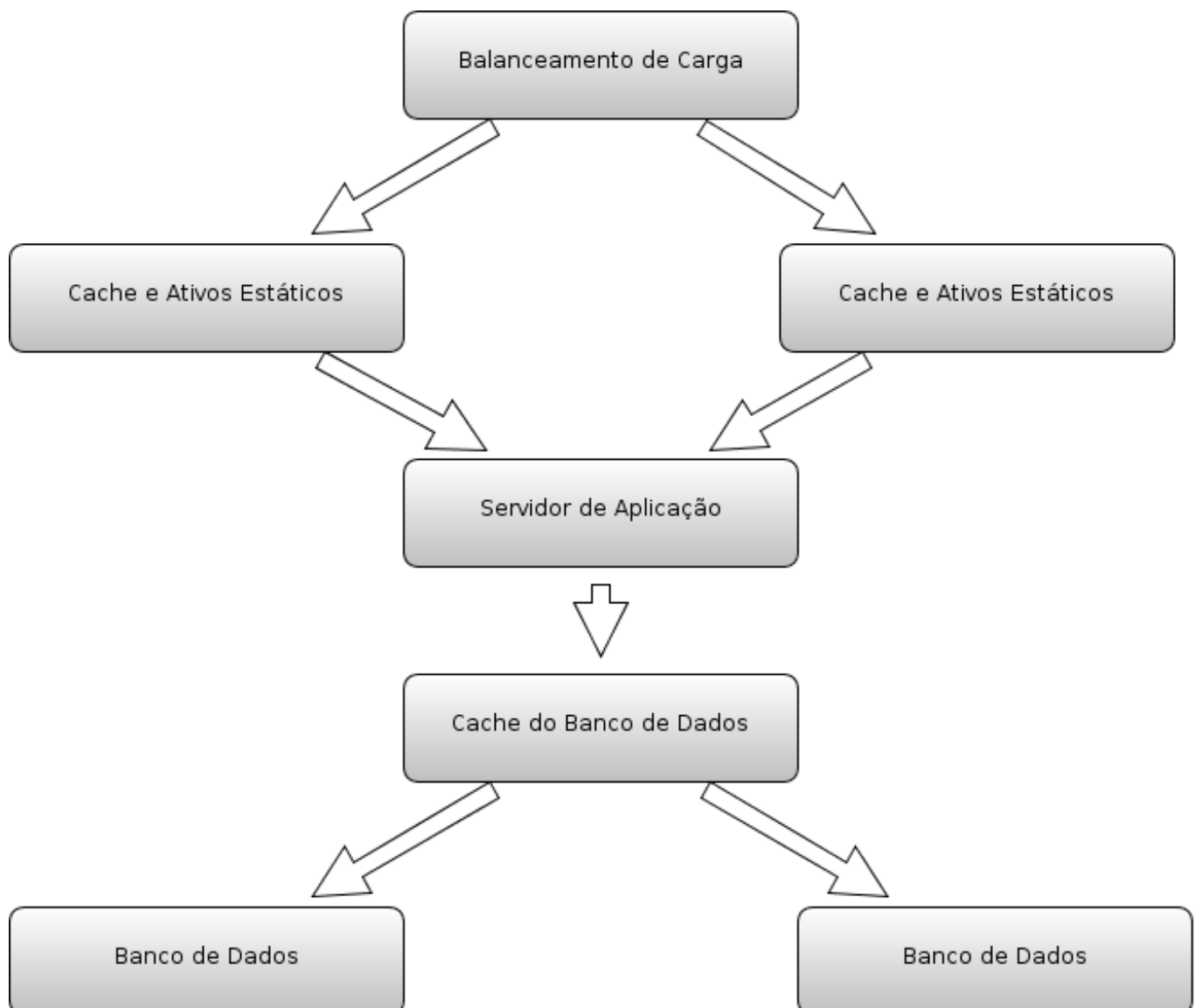


Melhorando a robustez de aplicações web

A Web tem crescido a uma taxa assustadoramente alta nos últimos anos, e agora possui mais de 2 bilhões de usuários. Os *Web sites* mais populares têm milhões de usuários. Devido a esses números, criar uma arquitetura para aplicações Web de larga escala, capazes de suportar milhões de usuários concorrentemente, é uma tarefa difícil, mas importante. Por exemplo, duas preocupações que um projetista tem que ter em mente são a latência (tempo de resposta) e a robustez do sistema.

Aqui, vamos nos preocupar com a robustez. Idealmente, uma aplicação Web deve ficar disponível 100% do tempo. Na prática, servidores falham, e é impossível criar uma arquitetura na qual haja uma garantia de uptime de 100%. Porém, é possível chegar arbitrariamente perto disso. Uma aplicação Web é normalmente estruturada em camadas. Cada camada possui alguma responsabilidade específica, e se comunica com as camadas adjacentes. A figura abaixo mostra um exemplo de arquitetura com 5 camadas:



Note que a arquitetura mostrada na figura possui problemas em potencial de robustez. Há apenas um servidor de aplicação. Se ele falhar, então o sistema não será capaz de atender nenhuma requisição que não possa ser servida diretamente do cache na camada 2. Uma maneira de contornar esse problema é adicionar um segundo servidor de aplicação. Assim,

mesmo que um deles falhe, ainda seria possível atender as requisições. Há, claro, a chance de que os dois servidores falhem ao mesmo tempo, mas ela é menor que a chance de que um único servidor falhe isoladamente. O mesmo vale para as outras camadas: é sempre possível adicionar mais servidores e melhorar a robustez daquela camada.

O problema, obviamente, é que servidores extras incorrem em custos extras. Os servidores em si possuem um custo alto, e ainda há o custo de manutenção e energia. Assim, ainda que ter, digamos, 400 servidores de aplicação pudesse aumentar muito a robustez do sistema, essa não seria uma solução interessante devido ao custo.

A probabilidade de falha da i -ésima camada, denotada por p_i , é definida como a probabilidade de que todos os servidores naquela camada falhem simultaneamente. Se a probabilidade de falha de um servidor individual naquela camada é f e há n servidores naquela camada, então

$$p_i = f^n$$

A robustez da camada i , denotada por r_i , é a probabilidade de que a camada i funcione, e é definida como

$$r_i = 1 - p_i = 1 - f^n$$

A robustez do sistema como um todo, denotada por R , é definida como a probabilidade de que todas as camadas do sistema funcionem, simultaneamente:

$$R = \prod_i r_i$$

Todos os servidores em uma mesma camada são idênticos. Em particular, eles possuem o mesmo custo, e a mesma probabilidade de falha. Porém, servidores de camadas diferentes podem ser diferentes.

São dados:

- o número de camadas N do sistema;
- o custo c_i de um servidor da i -ésima camada;
- a probabilidade de falha f_i de um servidor da i -ésima camada e
- o custo total máximo B .

Você deve determinar qual é a robustez máxima para o sistema como um todo (R) que é possível obter de forma tal que o custo não ultrapasse B .

Observações

Se uma camada tem zero servidores, então sua robustez é zero.

Entrada

Há vários casos de teste.

Cada caso de teste começa com uma linha contendo dois inteiros N e B , respectivamente o número de camadas no sistema ($1 \leq N \leq 100$) e o custo total máximo ($1 \leq B \leq 1000$) em milhares

de reais. Em seguida, há N linhas. A i-ésima dessas linhas possui dois números, c_i e f_i . c_i é um inteiro que representa o custo de um servidor na i-ésima camada em milhares de reais ($1 \leq c_i \leq 200$). f_i é um número de ponto flutuante que representa a probabilidade de falha de um servidor da i-ésima camada ($0 < f_i \leq 1$, o número é dado com 3 casas decimais de precisão).

A entrada termina com $N=B=0$, que não deve ser processado.

Saída

Para cada caso de teste, imprima uma linha contendo um único número, a robustez R máxima que é possível obter para o sistema descrito sem estourar o custo máximo. Esse valor deve ser impresso com 3 casas decimais de precisão.

Exemplos

Entrada:

```
3 105
30 0.100
15 0.200
20 0.500
0 0
```

Saída:

```
0.648
```

Nesse caso, a melhor opção é comprar um servidor para a primeira camada, dois servidores para a segunda camada e dois servidores para a terceira camada, a um custo total de $30 + 15 \cdot 2 + 20 \cdot 2 = 100$. A robustez da primeira camada será $1 - 0.1^1 = 0.9$. A da segunda camada será $1 - 0.2^2 = 0.96$ e a da terceira camada será $1 - 0.5^2 = 0.75$. Portanto, a robustez total é de $0.9 \cdot 0.96 \cdot 0.75 = 0.648$.