

Augusta – Aquecimento

1^a SELETIVA INTERNA – 2010/1

Sevidor BOCA:

`http://192.168.0.21:8001`
(acesso interno)

`http://200.19.107.16:8001`
(acesso externo)



Organizadores:

Alexandre Gonçalves Silva, Roberto Silvio Ubertino Rosso Jr., Claudio Cesar de Sá
{alexandre,rosso,claudio} at joinville dot udesc dot br

Lembretes:

- É permitido consultar livros, anotações ou qualquer outro material impresso durante a prova.
- A correção é automatizada, portanto, siga atentamente as exigências da tarefa quanto ao formato da entrada e saída de seu programa. Deve-se considerar entradas e saídas padrão.
- Procure resolver o problema de maneira eficiente. Se o tempo superar o limite pré-definido, a solução não é aceita. As soluções são testadas com outras entradas além das apresentadas como exemplo dos problemas.
- Teste seu programa antes de submetê-lo. A cada problema detectado (erro de compilação, erro em tempo de execução, solução incorreta, formatação imprecisa, tempo excedido ...), há penalização de 20 minutos. O tempo é critério de desempate entre duas ou mais equipes com a mesma quantidade de problemas resolvidos.
- Utilize o *clarification* para dúvidas da prova. Os juízes podem opcionalmente atendê-lo com respostas acessíveis a todos.

Problema A: Número de Fitromacci

Arquivo: `fitromacci.[c|cpp|java]`

Uma sequência de números é dada pela seguinte relação de recorrência: os k primeiros números são iguais a 1; o n -ésimo ($n > k$) valor é determinado pela soma dos k elementos anteriores.

Sua tarefa, neste exercício, será determinar o n -ésimo termo de Fitomate. Por exemplo, para $k = 3$ e $n = 7$, temos a sequência

1 1 1 3 5 9 17 ...

e, portanto, o sétimo número, neste caso, é 17.

Especificação da entrada

A primeira linha da entrada contém um inteiro T ($1 \leq T \leq 1000$) indicando o número de instâncias. Cada instância é composta por uma linha contendo os inteiros k ($1 \leq k \leq 7$), seguido por n ($1 \leq n \leq 40$).

Especificação da saída

Para cada instância, imprima, em um linha, o n -ésimo termo da sequência. Você pode assumir que o número de Fritromacci é um inteiro de 64 bits (em C/C++, o tipo é “long long” e, em JAVA, “long”).

Exemplo de entrada

```
3
3 7
7 20
```

Exemplo de saída

```
17
24097
```

Problema B: Bafo

Arquivo: `bafo.[c|cpp|java]`

Álbuns de figurinhas – sejam de times de futebol, princesas ou super-heróis – têm marcado gerações de crianças e adolescentes. Conseguir completar um álbum é uma tarefa muitas vezes árdua, envolvendo negociações com colegas para a troca de figurinhas. Mas a existência das figurinhas propicia uma outra brincadeira, que foi muito popular entre crianças no século passado: o jogo de bater figurinhas (o famoso “Bafo”). O jogo é muito simples, mas divertido (e muito competitivo). No início de uma partida, cada criança coloca em uma pilha um certo número de figurinhas. Uma partida é composta de rodadas; a cada rodada as crianças batem com a mão sobre a pilha de figurinhas, tentando virá-las com o vácuo formado pelo movimento da mão. As crianças jogam em turnos, até que a pilha de figurinhas esteja vazia. Ganha a partida a criança que conseguir virar mais figurinhas.

Aldo e Beto estão jogando bafo com todas as suas figurinhas e pediram sua ajuda para calcular quem é o vencedor. Você deve escrever um programa que, dada a quantidade de figurinhas que Aldo e Beto viraram em cada rodada, determine qual dos dois é o vencedor.

Especificação da entrada

A entrada é composta de vários casos de teste, cada um correspondendo a uma partida entre Aldo e Beto. A primeira linha de um caso de teste contém um número inteiro R que indica quantas rodadas ocorreram na partida. Cada uma das R linhas seguintes contém dois inteiros, A e B , que correspondem, respectivamente, ao número de figurinhas que Aldo e Beto conseguiram virar naquela rodada. Em todos os casos de teste há um único vencedor (ou seja, não ocorre empate). O final da entrada é indicado por $R = 0$. A entrada deve ser lida do dispositivo de entrada padrão. **Restrições:**

$1 \leq R \leq 1000$ ($R = 0$ apenas para indicar o final da entrada); $0 \leq A \leq 100$; $0 \leq B \leq 100$.

Especificação da saída

Para cada caso de teste da entrada, seu programa deve produzir três linhas na saída. A primeira linha deve conter um identificador do caso de teste, no formato “**Teste n** ”, onde n é numerado sequencialmente a partir de 1. A segunda linha deve conter o nome do vencedor (Aldo ou Beto). A terceira linha deve ser deixada em branco. A grafia mostrada no exemplo de saída, abaixo, deve ser seguida rigorosamente. A saída deve ser escrita no dispositivo de saída padrão.

Exemplo de entrada

```
2
1 5
2 3
3
0 0
4 7
10 0
0
```

Exemplo de saída

```
Teste 1
Beto

Teste 2
Aldo
```

Problema C: Geração de números pseudo-aleatórios

Arquivo: `pseudo.[c|cpp|java]`

John von Neumann sugeriu, em 1946, um método para criar uma sequência de números pseudo-aleatórios. Sua ideia é conhecida como método do “meio-quadrado” e funciona da seguinte maneira: Nós escolhemos um valor inicial a_0 , que tem uma representação decimal de comprimento, no máximo, n . Em seguida, multiplicamos o valor a_0 por si mesmo, e adicionamos zeros à esquerda até chegarmos a uma representação decimal de comprimento $2 \times n$, e tomamos os n dígitos do meio para formar a_i . Este processo é repetido para cada a_i , sendo $i > 0$. Neste problema, usamos $n = 4$.

Exemplo 1: $a_0 = 5555$, $a_0^2 = 30858025$, $a_1 = 8580$, ...

Exemplo 2: $a_0 = 1111$, $a_0^2 = 01234321$, $a_1 = 2343$, ...

Infelizmente, este gerador de números aleatórios não é muito bom quando começamos com um valor inicial que não produz todos os outros números com o mesmo número de dígitos.

Sua tarefa é verificar, para um dado valor inicial a_0 , quantos números diferentes são produzidos.

Especificação da entrada

A entrada contém vários casos de teste. Cada caso de teste consiste de uma linha contendo a_0 ($0 < a_0 < 10000$). Os números são possivelmente preenchidos com zeros à esquerda de tal forma que cada número seja composto por exatamente 4 dígitos. A entrada é terminada com uma linha contendo o valor 0.

Especificação da saída

Para cada caso de teste, imprima uma linha contendo o número de diferentes valores a_i produzidos por este gerador de números aleatórios, quando iniciado com o valor dado a_0 . Note que a_0 também deve ser contado.

Exemplo de entrada

```
5555
0815
6239
0
```

Exemplo de saída

```
32
17
111
```