

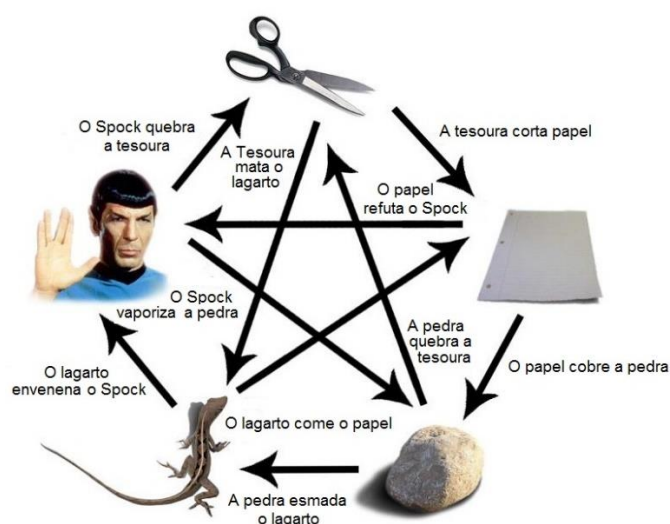
VII BXComp

7º Campeonato de Programação para Calouros do Curso de Sistemas de Informação 2017

7ª Etapa – Desafio 7 (5 Pontos)

Help Sheldon

Todos nós conhecemos e já brincamos de pedra, papel ou tesoura, mas o Sheldon, baseado na invenção de Sam Kass, propôs um bem diferente: pedra, papel, tesoura, Spock e lagarto. É bem simples: “tesoura corta papel, papel cobre pedra, pedra esmaga lagarto, lagarto envenena Spock, Spock dobra tesoura, tesoura decapita lagarto, lagarto come papel, papel refuta Spock, Spock vaporiza pedra e, como sempre, pedra esmaga tesoura”. O esquema pode ser melhor visualizado na figura a seguir:



Fonte: frontdaciencia.blogspot.com.br

Sheldon gostaria que você desenvolvesse um programa que determine o ganhador dentre os interessados em participar do jogo.

Tarefa

Seu programa deverá simular a execução de partidas do jogo para **N** participantes, numerados de **1** até **N**, e determinar qual dos jogadores é o ganhador, ou seja, aquele que vencer todos os oponentes em uma rodada. Todos os jogadores disputam entre si e suas jogadas são representadas por um inteiro **j** ($1 \leq j \leq 5$), associado da seguinte maneira: 1 = pedra, 2 = papel, 3 = tesoura, 4 = Spock e 5 = lagarto. Em caso de empate – mais de um ganhador por rodada, continuam participando apenas os jogadores que ganharam de pelo menos um outro jogador e os demais participantes serão desclassificados e terão suas jogadas representadas por zero nas rodadas subsequentes. Para as rodadas adicionais, os jogadores que continuam participando deverão trocar suas jogadas da seguinte maneira: soma-se ao valor **j** da jogada anterior, o valor **X** ($1 \leq X \leq N$) referente a numeração do jogador e subtrai **1**, então, o jogador **1**, por exemplo, nunca troca sua jogada, obtendo-se o novo valor **n** e caso **n** seja maior que cinco, deverá ser calculado o resto da divisão de **n** por cinco. O jogo deverá ser **encerrado** quando o número de rodadas extras for **igual ao número de jogadores**.

Entrada

Seu programa receberá, para cada caso de teste, uma linha contendo **N** inteiros ($2 \leq N \leq 1000$), que representam as jogadas iniciais dos **N** participantes. Cada jogada é representada por um inteiro **j** ($1 \leq j \leq 5$). O programa deverá parar de executar quando receber a entrada “exit”.

Saída

Seu programa exibirá na tela, para cada caso de teste, em uma linha a mensagem “O jogador **X** venceu!”, onde **X** ($1 \leq X \leq N$) é a posição do jogador que venceu a rodada. Nos casos de empate, o programa deve imprimir, com uma linha por rodada, os inteiros que representam as jogadas subsequentes dos jogadores que ainda estão participando e o valor 0 para os demais jogadores, separadas por um espaço em branco e sem espaço após a última jogada até que haja um ganhador. Caso o número de rodadas atinja o número de jogadores e nenhum jogador tenha ganhado até o momento, deverá ser impressa a mensagem “Ninguém ganhou!”. Após o último caso de teste deverá haver uma quebra de linha.

Exemplo de Entrada

```
1 1
1 2
4 2 2 1
3 1 2 3 2
1 1 1 3 3 4 4
exit
```

Exemplo de Saída

```
1 2
O jogador 2 venceu!
O jogador 2 venceu!
4 3 4 0
4 0 1 0
O jogador 1 venceu!
3 2 4 1 1
3 3 1 4 5
3 4 3 2 4
3 5 5 5 3
3 0 0 0 2
O jogador 1 venceu!
1 2 3 0 0 4 5
1 3 5 0 0 4 1
1 4 2 0 0 4 2
0 5 4 0 0 4 3
0 1 1 0 0 4 4
0 0 0 0 0 4 5
O jogador 7 venceu!
```