

Problema: Batalha entre tipos de civilizações

Você e seus amigos estão participando de um GameJam e vocês tem exatamente 24 horas para fazer um jogo! O jogo que vocês decidiram é um RPG onde o jogador escolhe uma civilização e deve proteger o seu povo contra a invasão da civilização inimiga. Existem três civilizações no jogo: a Tribo da Água, a Nação do Fogo e o Reino da Terra. Cada civilização possui vantagens e desvantagens no momento do combate:

- A Tribo da Água tem vantagem sobre a Nação do Fogo, mas tem desvantagem contra o Reino da Terra.
- A Nação do Fogo tem vantagem sobre o Reino da Terra, mas tem desvantagem contra a Tribo da Água.
- O Reino da Terra tem vantagem sobre a Tribo da Água, mas tem desvantagem contra a Nação do Fogo.

Cada amigo ficou responsável por uma parte da programação do jogo, enquanto você está responsável por calcular o dano que cada ataque possui. O dano é calculado a partir do do nível do jogador, que pode ir de 1 à 3, e o tipo da civilização, que pode ter vantagem ou desvantagem.

Definindo o multiplicador de nível

O multiplicador do nível do jogador é calculado seguindo a tabela a seguir:

Civilização	Multiplicador		
	Nível 1	Nível 2	Nível 3
Fogo	2.0	3.0	5.0
Água	1.0	2.5	4.0
Terra	2.5	5.5	7.0

Faça uma função para cada tipo de civilização que retorne, baseado no nível, qual o multiplicador de nível que um determinado ataque vai ter. Ou seja, crie as funções:

```
double fogo(int nivel);
double agua(int nivel);
double terra(int nivel);
```

Onde o primeiro argumento é um número inteiro nível que varia entre 1 à 3.

A função deve retornar o valor do multiplicador de acordo com o nível que foi passado.

Por exemplo, caso a função *agua* seja chamada da forma:

```
agua(2);
```

A função deve retornar o valor:

```
2.5
```

Como segundo exemplo, caso a função *fogo* seja chamada da forma:

```
fogo(2);
```

A função deve retornar o valor:

```
3.0
```

Definindo o multiplicador de tipo

O multiplicador de tipo é a vantagem ou desvantagem que o jogador recebe dependendo do tipo de civilização que ele escolheu e o tipo de civilização inimiga. A vantagem é multiplicar o dano por 2; a desvantagem é dividir o dano por 2. Se não há vantagem (o jogador está lutando contra a própria civilização), não há alterações no dano final.

Ou seja:

Jogador	CPU		
	Fogo	Água	Terra
Fogo	1.0	0.5	2.0
Água	2.0	1.0	0.5
Terra	0.5	2.0	1.0

Faça uma função que calcule a vantagem que uma civilização terá ao atacar outra civilização.

Para facilitar a identificação de cada civilização, considere que:

- Terra = 1
- Fogo = 2
- Água = 3.

Crie a função:

```
double calculaVantagem(int tipoPlayer, int tipoCPU);
```

Onde o primeiro argumento é o tipo da civilização do jogador, que varia entre 1 à 3, e o segundo argumento é o tipo de civilização do inimigo, que também varia de 1 à 3. A função deve retornar a vantagem que o jogador vai receber ao enfrentar a civilização do inimigo.

Por exemplo, caso a função seja chamada da forma:

```
calculaVantagem(1, 3);
```

A função deve retornar o valor:

```
2.0
```

Como segundo exemplo, caso a função seja chamada da forma:

```
calculaVantagem(2, 2);
```

A função deve retornar o valor:

```
1.0
```

Calculando o dano

Agora que temos os multiplicadores calculados, podemos utilizar as funções anteriores para calcular o dano. O dano é definido por:

$$100 \times multNivel \times multTipo - defesa = dano$$

Caso a conta resulte em um dano negativo, então ele deve ser desconsiderado.

Utilizando as funções anteriores, crie a função:

```
double calculaDano(int tipoPlayer, int nivelPlayer, int tipoCPU, double defesaCPU);
```

Onde:

- *tipoPlayer* é um número inteiro que indica o tipo da civilização do jogador, que varia entre 1 à 3;
- *nivelPlayer* é um número inteiro que indica o nível do jogador, que varia entre 1 à 3;
- *tipoCPU* é um número inteiro que indica o tipo da civilização do inimigo, que varia entre 1 à 3.
- *defesa* é um número real que indica a durabilidade da armadura do inimigo, protegendo-o do ataque, variando de 50 à 3000.

Que calcule a quantidade de dano que um determinado inimigo irá receber.

Por exemplo, caso a função seja chamada da forma:

```
calculaDano(1, 1, 2, 73.4);
```

A função deve retornar o valor:

```
51.600000
```

Como segundo exemplo, caso a função seja chamada da forma:

```
calculaDano(1, 3, 3, 1475.892);
```

A função deve retornar o valor:

```
0.000000
```

Atenção: Para este exercício, envie somente as funções *fogo*, *agua*, *terra*, *calculaVantagem* e *calculaDano* com a biblioteca *stdio.h*. Não inclua a função *main*.

Entrada

Não há dados de entrada para ser lidos.

Saída

Não há dados de saída para serem impressos.