

Algoritmo - Jogo

Agora que você já conhece comandos de fluxo, ou seja, os comandos da linguagem de programação que constroem nossos algoritmos, vamos tentar usar o que já estudamos para construir uma sequência de instruções que leva a um objetivo. Imagine que você quer construir um jogo em que o seu personagem caminha por uma estrada e a cada pedra que ele encontra, se for de ouro, vale 100 pontos, prata 50 pontos e normal não vale nada. Conforme ele anda, a cada 5 passos ele tem a chance de achar uma dessas três pedras. O personagem inicia no passo zero e ele pode andar até 1000 passos ou até juntar 500 pontos em pedras. Usando o que você aprendeu, desenvolva e apresente seu pseudocódigo correspondente à sequência de instruções que identifique corretamente a quantidade de passos e de pontos do início ao final do jogo.

Algoritmo Jogo

```
quantidadeDePontos = 0
```

```
listaDePossiveisPedras = [Ouro, Prata, Normal]
```

```
// Loop para rodar cada vez que o jogador dá um passo
```

```
PARA totalDePassos = 0; enquanto totalDePassos menor ou igual a 1000;  
totalDePassos++
```

```
    // Verifica se o número é divisível por 5
```

```
    SE totalDePassos dividido por 5 tem resto 0:
```

```
        pedra = listaDePossiveisPedras[numero aleatório]
```

```
        SE (pedra igual a Ouro)
```

```
            quantidadeDePontos += 100 pontos
```

```
        SE SENAO (pedra igual a Prata)
```

```
            quantidadeDePontos += 50 pontos
```

```
        SENAO
```

```
            quantidadeDePontos += 0 pontos
```

// Para de rodar o loop caso a quantidade de pontos seja maior ou igual a 500 pontos

SE quantidadeDePontos maior ou igual a 500 pontos

QuebrarLoopPara()

// Mostra a quantidade de pontos e o totalDePassos

ESCREVA quantidadeDePontos

ESCREVA totalDePassos