

Prática

Padrão Template Method

Considerando a prática anterior sobre Padrão Composite, em que temos:

“um método jogar(Personagem p) para simular um jogador andando pelo labirinto, de forma que a função seleciona aleatoriamente (random()) uma das duas possibilidades (caminhos) para prosseguir, até que o jogador encontre um nó folha (saída ou abismo). A medida que o jogador passa por uma sala, deve-se mostrar sua identificação, ou tipo, e o inimigo encontrado, assim como o *life* atual do personagem.”

Pede-se:

Transforme o método jogar() em um Template Method que utiliza as primitivas abstratas selecionarCaminho() e lutar(). As subclasses concretas do template devem implementar as primitivas abstratas de formas diferentes, conforme a seguir:

selecionarCaminho():

- uma implementação que considera uma seleção automática aleatoriamente do caminho a ser seguido;
- uma implementação que pergunta ao jogador (usuário) qual caminho ele deseja.

lutar():

- uma implementação que considera um valor randômico entre 1 a 10 para que o personagem sofra dano resultante do ataque do inimigo. Caso o personagem não morra (*life* fique maior que zero), ele pode aceder a próxima fase;
- uma implementação que gera um valor randômico entre 1 e 10 e pergunta ao jogador (usuário) qual valor foi gerado. Caso o jogador acerte, o inimigo é vencido sem que o personagem sofra nenhum dano. Caso contrário, o dano a ser aplicado é o valor fornecido pelo jogador (usuário). Caso o personagem não morra (*life* fique maior que zero), ele pode aceder a próxima fase.