

# Atividade para entrega

---

Apresentamos para consulta o exercício 7.27 do livro C – Como programar – Deitel – 6 ed. a seguir. Este exercício é uma generalização dos exercícios 7.25 e 7.26 da mesma referência. Pede-se

**7.25**      **7.25 (Travessia de Labirinto)** A grade de uns e zeros a seguir é uma representação bidimensional de um labirinto.

```
111111111111 100010000001 001010111101 111010000101 100001110100 111101010101
100101010101 110101010101 100000000101 111111011101 100000010001
111111111111
```

Os uns representam as paredes do labirinto, e os zeros representam os quadrados (espaços) dos quais é possível cruzar o labirinto.

Há um algoritmo simples para atravessar um labirinto que garante a localização da saída (admitindo que há uma saída). Se não houver uma saída, você chegará novamente ao local de partida. Coloque sua mão direita na parede de sua direita e comece a andar para a frente. Nunca tire sua mão direita da parede. Se o labirinto dobrar para a direita, siga a parede da direita. Contanto que sua mão direita não seja retirada da parede, você acabará chegando à saída. Pode haver um caminho mais curto, mas esse garante **sua saída do labirinto**. Escreva uma função

---

recursiva `travessiaLabirinto` para atravessar um labirinto. A função deve receber como argumentos um array 12-por-12 de caracteres representando o labirinto e um local de **p**: A medida que a função `travessiaLabirinto` tentar localizar uma saída do labirinto, ela deve colocar um caractere **X** em cada quadrado do caminho percorrido. A função deve exibir o labirinto depois de cada movimento para que o usuário possa ver como sua saída será encontrada.

**7.26**      **(Gerando Labirintos Aleatoriamente)** Escreva uma função `geradorLabirinto` que tome um array 12-por-12 de caracteres como argumento e produza aleatoriamente um labirinto. A função também deve fornecer os locais de partida e de saída do labirinto. Experimente sua função `travessiaLabirinto` do Exercício 7.25 usando vários labirintos gerados aleatoriamente.

**7.27**      **(Labirintos de Qualquer Tamanho)** Generalize as funções **`travessiaLabirinto`** e **`geradorLabirinto`** dos Exercícios 7.25 e 7.26 para processar labirintos de qualquer largura e altura.

Pede-se

- Desenvolver as funções solicitadas em 7.27
- Apresentar o labirinto usando a `glut` e a `OpenGL`, conforme ilustra o exemplo em Fig. a
- Apresentar uma animação da travessia do labirinto, conforme ilustra o exemplo em Fig. b

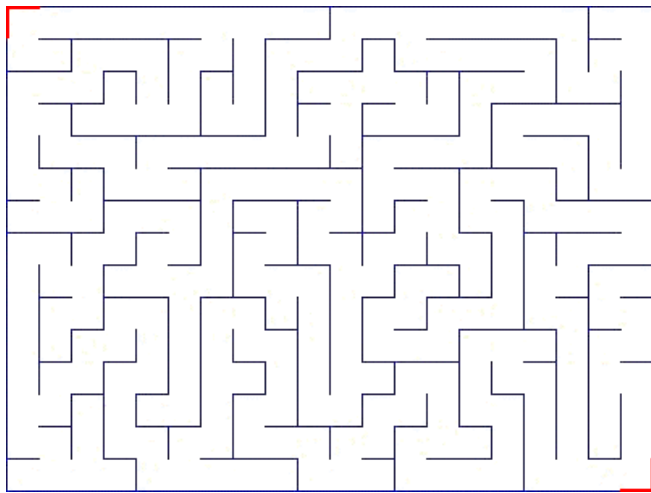


Fig.a Labirinto

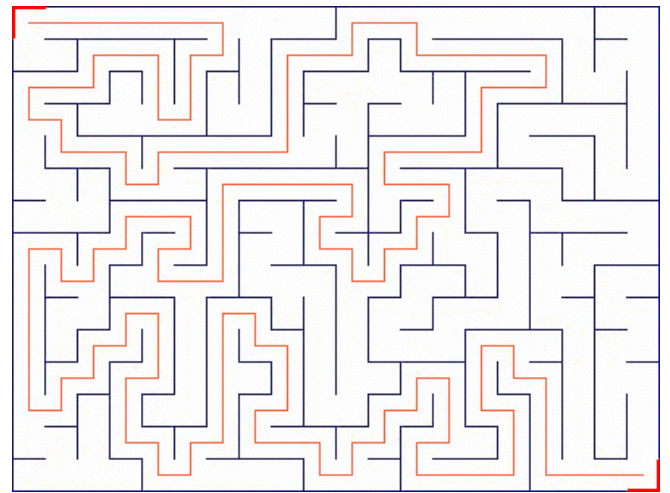


Fig.b Caminho Percorrido