

Laboratório 12: Labirinto

Segundo Semestre de 2017 - Turmas Coordenadas

Peso da Atividade: 2

Prazo de Entrega: 12 de novembro de 2017 às 23:59:59

Conteúdo

[Contexto](#)

[Tarefa](#)

[Observações da Tarefa](#)

[Exemplos](#)

[Observações Gerais](#)

[Critérios Importantes](#)

Contexto

Se ela fosse por aquele lado, iria direto para o castelo!

— Verme no filme Labirinto, 1986



Figura 1: Goblins tomando conta de Toby.

Você foi deixada sozinha em casa para cuidar de Toby, seu irmãozinho e filho da sua madrasta. Irritada com o choro dele, você desejou em voz alta que o rei dos duendes o levasse. E, por mais que você não acreditasse em duendes, agora ele apareceu e levou Toby para seu castelo no centro do labirinto.

- Volte para o seu quarto — diz o rei dos duendes — Brinque com seus brinquedos e fantasias. Esqueça o bebê!
- Não posso, não entende que não posso?
- Desista! Meu labirinto é muito difícil e você nunca conseguirá chegar até seu irmãozinho, veja!

Você observa o labirinto. O castelo fica no centro. Em torno dele, estão **n** muralhas circulares altas, uma dentro da outra, formando grandes corredores circulares. Há passagens em pontos específicos das muralhas, mas também há barreiras nos corredores impedindo que se dê a volta completa.

— Desista! Não há mais do que um caminho entre dois lugares no meu labirinto. E vou dar apenas 13 horas pra você chegar ao centro, antes que seu lindo irmãozinho se transforme em um de nós, para sempre! Mesmo que você tenha visto todo o meu labirinto, como você poderá navegar lá dentro?

Você não perde tempo e inicia a sua jornada. Pretende valer-se do sol e da visão do alto castelo para determinar os ângulos percorridos nos corredores e assim conseguir realizar o caminho planejado.

Baseado no filme [Labirinto](#) - [A magia do tempo](#).

Tarefa

Sua tarefa é determinar o caminho a percorrer no labirinto para chegar ao castelo no seu centro e resgatar Toby. Seu programa receberá na primeira linha o número `n` das muralhas. As $(2n - 1)$ linhas seguintes são, alternadamente, uma linha de passagens de uma muralha e uma linha de barreiras de um corredor. Dos mais externos ao mais interno. Na muralha mais externa só há uma passagem. Como as passagens são através de muralhas circulares e as barreiras bloqueiam corredores circulares, a posição das passagens e barreiras será dada em graus, ao redor do castelo. Os valores de cada linha serão números inteiros entre 0 e 359, ordenados.

Para representar o caminho encontrado, a saída é composta pela seqüência de passagens que se deve atravessar para ir do lado de fora da muralha mais externa para o lado de dentro da muralha mais interna. Deve ser indicada uma passagem por linha. Note que você pode usar uma passagem indo do anel mais externo para o mais interno ou do mais interno para o mais externo. Por isso, antes de indicar a posição da passagem (em graus), você deve imprimir 'E' se ela for usada para entrar do mais externo para o mais interno e 'S', caso contrário. Observe os exemplos, em particular o Exemplo 3, e suas ilustrações.

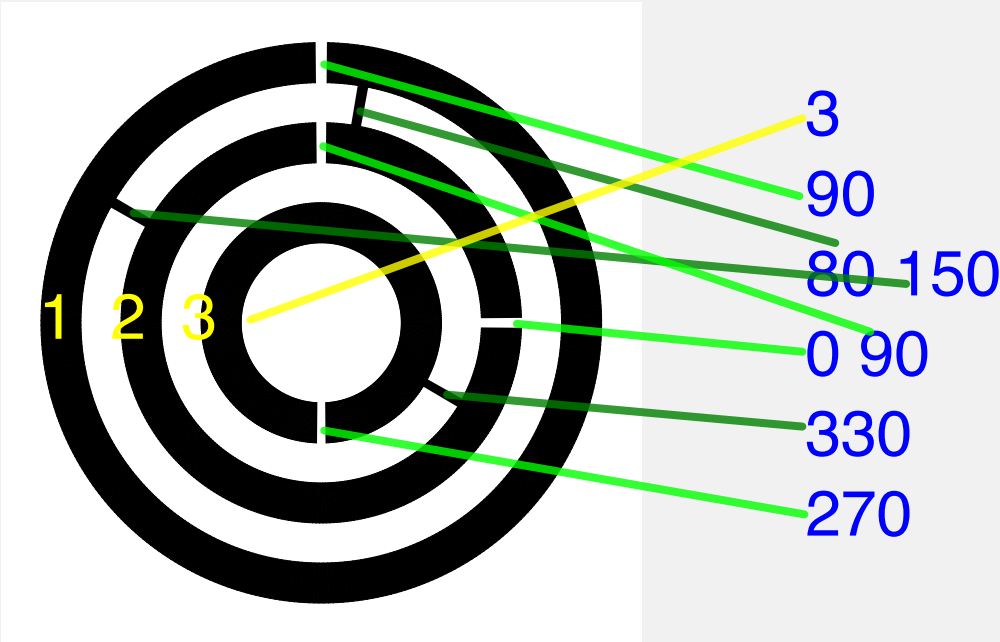


Figura 2: Exemplo de entrada. Para a saída veja o exemplo de execução 1.

Observações da Tarefa

- Serão, no máximo, dez muralhas a atravessar.
- Os dois lados da passagem (para o anel mais interno e mais externo) estão sempre livres de barreiras.
- Após cada impressão, indicando uma instrução de caminho, deve haver um `\n`.
- Você deve implementar e utilizar uma função recursiva para resolver o problema.

Exemplos

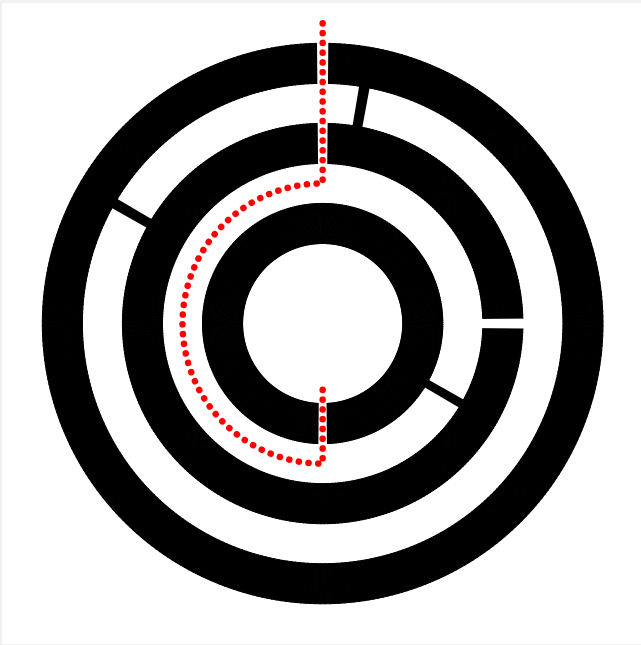
Notas:

Textos em azul designam dados de entrada, isto é, que devem ser lidos pelo seu programa.
Textos em preto designam dados de saída, ou seja, que devem ser impressos pelo seu programa.
Nas figuras, a saída está representada pela linha pontilhada vermelha.

Exemplo de execução 1:

```
3
90
80 150
0 90
330
270

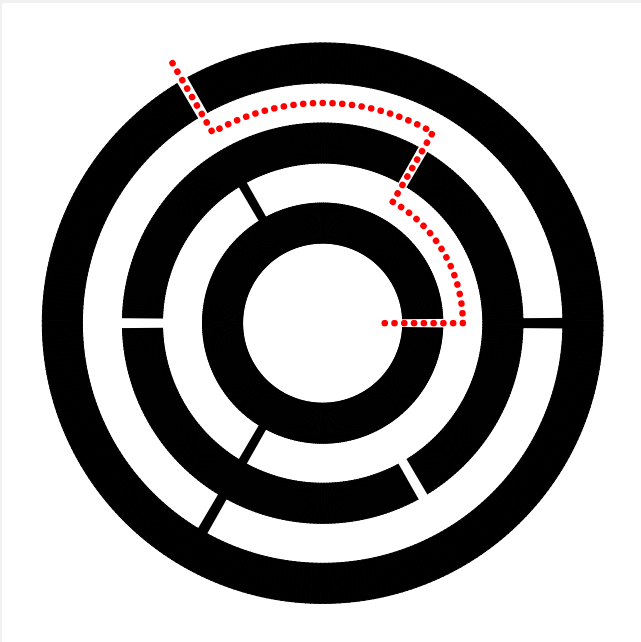
E 90
E 90
E 270
```



Exemplo de execução 2:

```
3
120
0 240
60 180 300
120 240
0
```

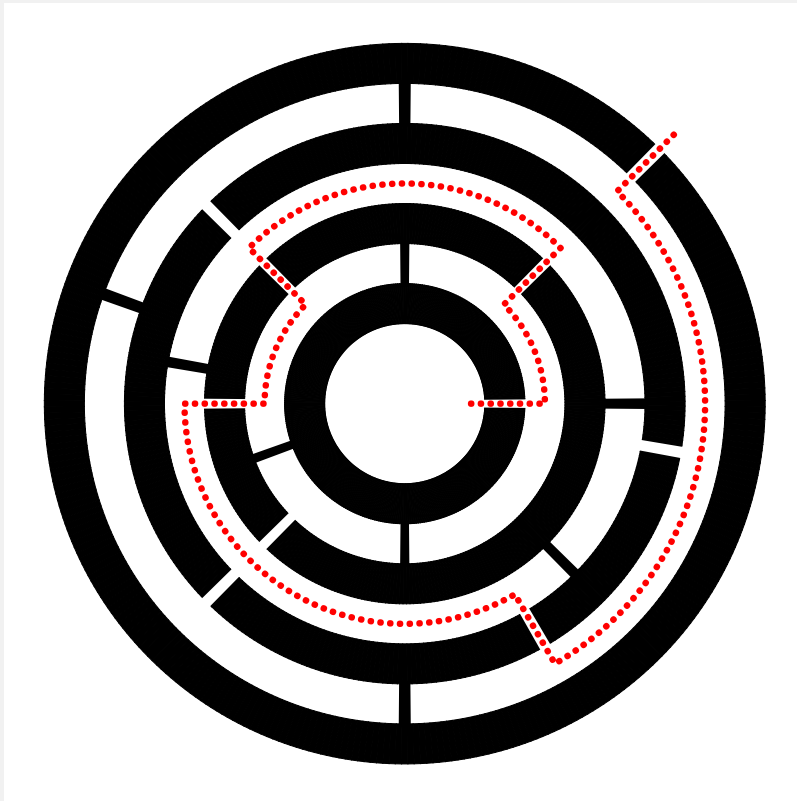
```
E 120
E 60
E 0
```



Exemplo de execução 3:

```
4
45
90 160 270
135 225 300 350
0 170 315
45 135 180 225
90 200 270
0
```

```
E 45
E 300
E 180
S 135
```



Observações Gerais

- O número máximo de submissões é 20.
- O arquivo `lab12.c` deve conter todo o seu programa.
- Para a realização dos testes automáticos, a compilação se dará da seguinte forma: `gcc lab12.c -o lab12 -Wall -Werror -ansi -pedantic`.
- Não se esqueça de incluir no início do programa uma breve descrição dos objetivos, da entrada, da saída, seu nome, RA e turma.
- Após cada submissão, você deve aguardar um minuto até poder submeter seu trabalho novamente.
- Ao final deste laboratório, você terá aprendido como utilizar funções recursivas.

CrITÉrios Importantes

O **não** cumprimento dos critérios abaixo acarretará em **nota zero na atividade**, independentemente dos resultados dos testes do SuSy.

- Sua solução deve atender todos os requisitos definidos no enunciado.
- Não serão aceitas soluções contendo estruturas não vistas em sala (para este laboratório, poderão ser utilizadas apenas variáveis simples, vetores, matrizes, operações de entrada e saída, operações aritméticas, desvios condicionais, strings, funções, apontadores, *structs* e recursão).
- Não é permitido o uso de `continue` e `break` (exceto em estruturas do tipo *switch-case*).
- Não é permitido o uso de variáveis globais.
- Seu programa deve conter uma chamada a uma função recursiva.
- O único cabeçalho aceito para inclusão é `stdio.h`.

