

Aula 9 - 15/09 - Mais Exercícios

Último Ensaio Antes da P1

“O que eu ouço eu esqueço, o que eu vejo eu lembro e o que eu faço eu sei.”
(Confúcio)

Inspiremo-nos nessa frase para nos motivar a **fazer** os exercícios abaixo.

Exercício 1: Escreva uma função recursiva que receba dois inteiros positivos k e n e calcule o valor de n^k . Quantas chamadas recursivas faz a sua função? É possível reduzir o número de chamadas recursivas? Se não, justifique. Se sim, reescreva sua função de forma que o número de chamadas recursivas seja significativamente reduzido.

Exercício 2: Escreva uma função recursiva que inverte uma lista ligada simples.

Exercício 3: Neste exercício, sua tarefa será projetar uma pista de corrida inspirada nos circuitos ovais de Fórmula Indy. Serão dadas n coordenadas no plano cartesiano. A primeira coordenada, denotada por P_0 , possui o menor valor no eixo y e representa o ponto de largada e chegada. As demais coordenadas serão dadas em ordem crescente com relação ao ângulo formado ao redor de P_0 (assuma que não há pontos colineares). Deseja-se que o circuito seja desenhado como um polígono cujos vértices são alguns dos pontos dados na entrada. Partindo da largada em sentido anti-horário, o circuito deve possuir apenas curvas à esquerda. Para atender a essa restrição, possivelmente alguns pontos dados na entrada deverão ser removidos do circuito, mas tais pontos deverão estar contidos no interior do “contorno” do circuito. **Dica:** Utilize a função boolean `curvaEsq(Ponto a, Ponto b, Ponto c)` `{return (b.x-a.x)*(c.y-a.y)-(b.y-a.y)*(c.x-a.x)>0.0;}` para determinar se passando pelos vértices **a**, **b** e **c**, nessa ordem, temos uma curva à esquerda (a função devolve **true**) ou à direita (a função devolve **false**). Exemplo de entrada e a respectiva saída:

Entrada	Saída
8	5
-1.0 0.0	-1.0 0.0
0.0 2.0	0.0 2.0
5.0 4.0	5.0 4.0
2.0 2.5	-2.0 4.0
1.0 1.0	-2.0 0.0
0.0 3.0	
-2.0 4.0	
-2.0 0.0	