

## Instruções condicionais

### Tópicos

- Expressões lógicas
- Instruções condicionais

### Exercícios

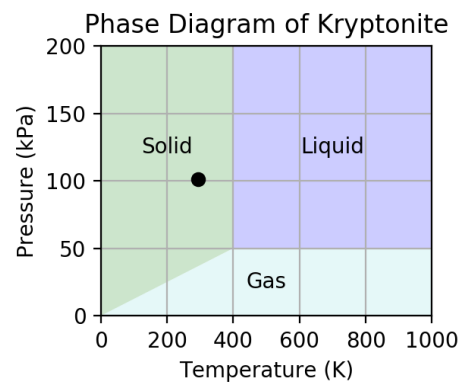
1. Responda a estes [exercícios sobre expressões lógicas e instruções condicionais](#). (O texto está em Inglês. Se precisar, use tradução automática.)
2. O programa `max2.py` lê dois números e mostra o maior deles. Teste-o e analise-o. Altere o programa para fazer o mesmo, mas usando uma instrução condicional em vez da função pré-definida `max`.
3. Copie o programa anterior para `max3.py` e modifique esta cópia para achar o maior de três números, sem usar a função `max`. Consegue fazê-lo só com duas comparações? E consegue achar o maior de quatro números só com três comparações?
4. Escreva um programa que leia um número inteiro e mostre uma mensagem indicando se o número é par ou ímpar. *Sugestão: use o operador `%` para obter o resto da divisão por 2.*
5. Um certo tarifário de telefone cobra 0.12 € por minuto, cobrado ao segundo após o primeiro minuto. Por exemplo, chamadas de 10 s, de 30 s ou de 60 s custam 0.12 €. Uma chamada de 90 s custa 0.12 (pelo primeiro minuto) + 0.06 (pelos restantes 30 s) = 0.18 €. Faça um programa que calcule o custo de uma chamada sabendo a duração em segundos.
6. Execute o programa `age.py` e teste-o com vários valores de entrada. Analise o código-fonte para perceber como funciona. Consegue detetar o erro semântico? Que valores de `age` produzem a categoria "grown-up"? Corrija o erro. Consegue reescrever o código com uma instrução `if-elif-else`? Tente simplificar o código para eliminar condições redundantes.
7. O índice de massa corporal (IMC) é uma medida usada para avaliar se uma pessoa tem um peso saudável para a sua estatura. O programa `imc.py` determina o IMC do utilizador e classifica-o apenas em duas categorias. Experimente o programa e analise-o. Modifique-o para dar uma de 4 categorias de acordo com a tabela abaixo.<sup>1</sup> Evite testar condições redundantes no programa.

<b>IMC (kg/m²):</b>	<18.5	[18.5, 25[	[25, 30[	30 ou mais
<b>Categoria:</b>	Magro	Saudável	Forte	Obeso

<sup>1</sup> Esta tabela não é oficial.

8. O diagrama de fases da Kryptonite (substância fictícia) é muito invulgar (ver figura ao lado). Diz-nos que em condições normais de temperatura e pressão (ponto assinalado na figura), a Kryptonite é sólida, mas se a temperatura for superior a 400 Kelvin e a pressão estiver acima de 50 kPa, passa à fase líquida. Também pode passar à fase gasosa se a pressão for baixa e a temperatura suficientemente elevada.

O programa `kryptonite.py` pede a temperatura e a pressão e deveria indicar a fase correspondente, mas tem vários erros.



- Execute o programa e veja o erro sintático reportado. Detete-o e corrija-o. *Sugestão: atenção aos parênteses.*
  - Volte a executar. Ocorre um `TypeError`? Porquê? Corrija.
  - Agora deve conseguir executar o programa, mas ainda tem erros semânticos. Por exemplo, a 300 K e 100 kPa a fase devia ser SOLID. Modifique as instruções condicionais para determinar a fase corretamente. *Sugestão: comece por determinar as equações das retas que dividem o espaço.*
  - Ajuste a string de formato do resultado para a temperatura aparecer com 1 casa decimal e a pressão com 3. O programa `examplesformat.py` tem vários exemplos de formatação.
9. Escreva um programa que peça as notas de duas componentes de avaliação, CTP e CP, e calcule a nota final (inteira) de um aluno de Fundamentos de Programação. Se alguma componente for inferior à nota mínima, a nota final deve ser 66 (código usado para indicar Reprovado por Nota Mínima). Se o aluno reprovar por nota mínima ou nota inferior a 10, o programa deve pedir as notas de recurso, ATPR e APR, e calcular a nova nota final. Consulte os pesos e regras nas [informações da disciplina](#).
10. Desafio: Um dardo atinge o alvo nas coordenadas (x, y). Complete o programa `darts.py` para mostrar a pontuação obtida. Procure mais [informações sobre o jogo](#) e sobre as [dimensões do alvo](#). Use o [fórum](#) para discutir o problema.

