

0. Desenvolva um algoritmo que leia um número e o transforme em binário. É fácil encontrar este algoritmo pronto na Internet; no entanto, lembre-se de que você nada ganharia com isso (esta lista não "vale nota", e ver o algoritmo pronto só elimina a sua oportunidade de aprender a desenvolvê-lo sozinho/a).

1. Refaça o algoritmo que calcula dias de vida usando funções.

2. (Fácil) Descreva um algoritmo que leia duas strings representando dois DNAs obtidos de um sequenciador automático. As posições da string podem conter A, C, G ou T (apenas). O algoritmo deve se recusar a analisar os DNAs se eles tiverem comprimento diferente. Se tiverem o mesmo comprimento, deve dizer quão similares eles são (verifique apenas se cada posição de um é igual à mesma posição do outro, e diga a porcentagem de posições similares).

3. Faça um programa que leia o raio e desenhe uma CIRCUNFERÊNCIA (e não um círculo).

Raio?

5

```
  * * * * *
 *           *
 *         * *
 *       *   *
 *     *     *
 *   *       *
 * *         *
 * *       *
 * *     *
 * *   *
 * * * * *
```

(Como dá pra notar, ela não ficará muito bonita. Com raio maior o desenho fica um pouco melhor...)

4. (ESTE É UM DAQUELES QUE VOCÊ NÃO PRECISA FAZER SE NÃO CONSEGUIR)

Faça um algoritmo que leia duas strings representando fitas de DNA, mas desta vez algumas posições podem conter não apenas A, C, T ou G. Podem conter:

```
R = G A (purina)
Y = T C (pirimidina)
K = G T (ceto)
M = A C (amino)
```

(Representando ambiguidade)

O algoritmo deve informar:

- i - Quanto os DNAs seriam similares se, cada vez que houver ambiguidade, considerarmos que as posições não casam;
- ii - Quanto os DNAs seriam similares se, cada vez que houver ambiguidade, considerarmos que as posições casam (se for possível).

Por exemplo, 'R' com 'A' contaria como não em (i), mas como sim em (ii).

'R' com 'Y' contaria sempre como não.