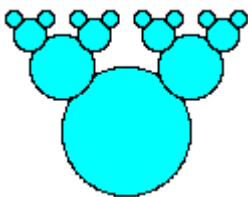


**Recursividade.****Exercícios**

1. O programa `recErrors.py` define duas funções recursivas, mas ambas estão erradas: não terminam. Em cada caso, procure detetar a causa e corrija o erro. *Sugestão: confirme se as funções satisfazem as condições de terminação discutidas na aula TP.*
2. No programa `genWords.py`, a função `genWords3(symbols)` permite gerar uma lista com todas as palavras de comprimento 3 que se podem formar com caracteres escolhidos de um alfabeto de símbolos dado. Escreva uma versão generalizada, `genWords(symbols, n)`, que permita gerar todas as palavras de comprimento `n`. Um algoritmo recursivo para obter cada uma das palavras de tamanho `n` consiste em obter a lista de todas as palavras de tamanho `n-1` e a cada uma delas acrescentar cada um dos símbolos do alfabeto. Qual será o caso base? E que resultado lhe corresponde?
3. O programa `findFiles.py` define uma função `printDirFiles(d)` que mostra o conteúdo de um diretório. Analise-a para recordar as funções que permitem listar diretórios e manipular nomes de ficheiros (*paths*). Complete a função `findFiles` para devolver uma lista com os nomes de todos ficheiros que têm uma certa extensão num dado diretório. A procura deve percorrer o diretório e todos os seus subdiretórios recursivamente.
4. Mover as letras 'x' para o fim de uma string. ([Codecheck](#))
5. Inverter os dígitos de um inteiro usando uma função auxiliar recursiva. ([Codecheck](#))
6. Extrair todas as strings de objetos compostos. ([Codecheck](#))
7. Mostrar uma representação explodida de objetos compostos. ([Codecheck](#))
8. Escreva uma função recursiva para criar figuras como esta. ([Codecheck](#))



4

<https://codecheck.io/private/resume/2201231057bp2sbaytmsrp8u4fh3w7gflqt/ebur-xawy-hini-emof/CEKJ88YQA50VXO3Q97BIFHT80>

5

<https://codecheck.io/private/resume/220123104237q5dz2twvxq43ite9kdz8zx1/joqu-tywy-irog-cilo/3KCZHA3PZL1DD6K>