# **TRABALHO PRÁTICO II**

# **Algoritmos Avançados de Bioinformática**

Neste trabalho prático é esperado que apliques os conhecimentos sobre variações de sequências e árvores de sufixo explorados nas aulas de Algoritmos Avançados de Bioinformática.

1. Com base no ficheiro gerado na aula 5 (HG00154.chr21.raw.vcf) cria scripts em Python que permitam responder às seguintes questões:
   1. Quantas mutações únicas podem ser identificadas no ficheiro (ex: se a mutação A>T ocorrer mais do que uma vez na sequencia só deverá ser contada uma).

*int perg\_a (fileName)*

* 1. Cria um gráfico de barras que conta as mutações SNP para cada um dos nucleótidos (ex: A>C, A>T, A>G).

*perg\_b(fileName)*

* 1. Calcula o número de SNPs, deleções e inserções.

*(int, int, int) perg\_c (fileName)*

1. Considere a implementação de árvores de sufixos desenvolvida na aula anterior dada pela classe SuffixTrie.
   1. Implemente uma função *compact(self)* que faça a compactação da árvore (self) de acordo com o indicado nos slides. Neste caso, considere que a chave do dicionário que representa os ramos é uma string que poderá ter mais do que um caractere (embora nos slides estejam representados índices, no nosso caso considerar-se-ão sempre as strings correspondentes).
   2. Adapte a função *findPattern* dada na classe de forma a poder funcionar com árvores compactadas (de acordo com o ex. 1)
   3. Implemente uma função – *repeats(self, k, ocs)* - que dada uma árvore de sufixos que representa uma string (vamos designar por genoma), retorne a lista de padrões com tamanho k, que ocorram no genoma um número de vezes maior ou igual a ocs.