

Avaliação contínua de Algoritmos Avançados de Bioinformática

3ª submissão enunciado e tópicos

O objetivo será que os grupos aprofundem uma das temáticas abordadas nas aulas, através do desenvolvimento de código Python.

Este trabalho dará liberdade aos alunos de poder explorar cada tópico de forma livre, escolhendo as melhores funcionalidades a implementar no código a desenvolver no âmbito genérico do tópico definido.

Os alunos deverão informar o docente por email dos elementos do grupo (2 ou 3) e da escolha do tópico, sendo este confirmado pelo docente (não poderão haver mais do que 2 grupos em cada tópico). O código final desenvolvido e um curto relatório (máximo 2 páginas), deverão ser submetidos no site de e-learning até ao dia **8 de junho de 2015**. O relatório deverá sumariar a abordagem seguida e as funcionalidades do código. Documentação mais técnica poderá ser incluída nos comentários dos próprios ficheiros de código.

Os tópicos possíveis a abordar serão:

Tema 1:

Escolha de uma biblioteca de grafos em python (e.g. NetworkX - <http://networkx.github.io/>, igraph - <http://igraph.org/python/>) e implementação sobre essa biblioteca de funções que permitam criar redes metabólicas (ou outro tipo de redes biológicas) e realizar a sua análise topológica. Poderá implementar algoritmos semelhantes aos implementados na aula, podendo complementar o código com outras funcionalidades de análise topológica que façam sentido no tipo de rede selecionado (ver artigo da 2ª aula). Poderá instalar a aplicação Cytoscape como fonte de inspiração.

Tema 2:

Escolha de uma biblioteca de algoritmos evolucionários em python (e.g. DEAP - <http://deap.readthedocs.org/en/master/>, PyGene ou PyEvolve -

<http://pyevolve.sourceforge.net/>) e implementação sobre essa biblioteca de código para o problema de descoberta de motivos, ou outro problema de otimização em bioinformática à escolha do grupo.

Tema 3:

Aprofundamento do código desenvolvido nas classes de grafos desenvolvidas nas aulas, podendo adicionar funções para diferentes tipos de análise topológica, algoritmos diversos sobre o grafo com significado biológico, criação de diferentes tipos de redes (e.g. regulatórias), criação de redes de várias fontes (e.g. bases de dados como KEGG, MetaCyc, ...) de forma automatizada, etc. Poderá instalar a aplicação Cytoscape como fonte de inspiração.

Tema 4:

Aprofundamento do código desenvolvido nas aulas para algoritmos de assembly (grafos de sobreposição e/ ou de deBruijn, circuitos eulerianos e hamiltonianos), podendo adicionar funções para tornar o código mais robusto para dados em maior escala, adicionar suporte a leituras emparelhadas, lidar com erros e problemas de cobertura, etc. Aconselha-se a visualização de algumas sessões sobre esta temática no curso “Bioinformatics Algorithms I” do coursera.

Tema 5:

Aprofundamento do código desenvolvido nas aulas para algoritmos de procura de padrões/ alinhamentos contra referências, podendo adicionar funções para tornar o código mais robusto para dados em maior escala, implementar árvores de sufixos com compactação, arrays de sufixos (completos e parciais, etc). Aconselha-se a visualização de algumas sessões sobre esta temática no curso “Bioinformatics Algorithms II” do coursera.

Tema 6:

Aprofundamento do código desenvolvido nas aulas para algoritmos evolucionários, podendo aumentar-se as capacidades do mesmo em termos de outras representações, operadores ou algoritmos, bem como abordar-se outros problemas, preferencialmente relacionados com bioinformática.