

# INF01124 - Classificação e Pesquisa de Dados - Laboratório 1

Entrega: 08/09/2020 pelo Moodle

Importante: Esta tarefa pode ser feita em duplas!

## 1 ShellSort

Implemente o algoritmo ShellSort com a sequência dada por  $\left\lfloor \frac{N}{2^k} \right\rfloor$ , onde  $N$  é o tamanho da lista. Para o exemplo abaixo, onde  $N = 9$ . A sequência fica 4, 2 e 1.

54 26 93 17 77 31 44 55 20

After increments of size 4 the list is [20, 26, 44, 17, 54, 31, 93, 55, 77]

After increments of size 2 the list is [20, 17, 44, 26, 54, 31, 77, 55, 93]

After increments of size 1 the list is [17, 20, 26, 31, 44, 54, 55, 77, 93]

Teste sua implementação nas duas listas abaixo. A saída do programa deve mostrar o estado da sequência após cada iteração, exatamente como mostrado acima. Envie o código e a saída do programa para correção.

A) 733 77 611 51 496 990 61 175 127 794 794 76 197 141 863

B) 872 424 729 573 877 696 321 758 949 217 767 238 562 175 999 12 380 406 225 995

## 2 Ciura, 2001

Implemente a sequência proposta por Ciura em [1] (<https://oeis.org/A102549>) e cronometre o tempo de execução em listas de inteiros aleatórios de tamanho 100, 1000, 10000 e 100000. A sequência de Ciura apresenta a melhor performance da literatura em relação ao número médio de comparações. Faça o mesmo para a sequência usada no exercício anterior e gere um gráfico de linha mostrando as performances das duas sequências. Não é necessário gerar a saída como no exercício anterior.

A sequência de Ciura = 1, 4, 10, 23, 57, 132, 301, 701

Para números maiores que 701, use a formula  $h_k = \lfloor 2.25h_{k-1} \rfloor$ .

## 3 Desafio Bônus - DNA Sorting

Resolva o problema "DNA Sorting"

[https://onlinejudge.org/index.php?option=com\\_onlinejudge&Itemid=8&category=8&page=show\\_problem&problem=553](https://onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&category=8&page=show_problem&problem=553)

Este é um problema de maratona de programação. O problema será considerado completo se for aceito no site da <https://uva.onlinejudge.org> e o código for entregue com o resto dos exercícios.

## Referências

- [1] Ciura, Marcin (2001). "Best Increments for the Average Case of Shellsort". In Freiwalds, Rusins. Proceedings of the 13th International Symposium on Fundamentals of Computation Theory. London: Springer-Verlag. pp. 106–117. ISBN 978-3-540-42487-1.