



UFOP

UNIVERSIDADE FEDERAL DE OURO PRETO

Rafael Francisco de Oliveira - 2021.10171

PCC104 - Projeto e Análise de Algoritmos

Github: raffoliveira

Lista 2



Execução das questões práticas

A estrutura da pasta denominada `lista_2` no GitHub segue abaixo:

A implementação dos algoritmos foi dividida em módulos. A pasta `exercicios` contém os arquivos de cada exercício. O arquivo chamado `functions.h` é responsável por ter as declarações de todas as funções enquanto o arquivo chamado `functions.cpp` contém as implementações de cada função. No momento da execução de cada algoritmo é necessário executar também o arquivo `functions.cpp`.

A pasta `testes` contém os arquivos contendo as instâncias para testes. Para a escolha de cada arquivo, basta modificar as linhas 30 e 31 do arquivo `functions.cpp`, as quais especificam o caminho do diretório da pasta e o nome do arquivo escolhido.

O ambiente de execução utilizado durante a criação dos algoritmos foi *Linux*. Por isso, para a execução dos algoritmos, basta executar o seguinte comando abaixo no terminal:

```
g++ exercicio_1.cpp functions.cpp -o exe  
./main
```

Questões teóricas

1. Apresente uma descrição da classe *vector* apresentando o custo computacional de cada uma de suas operações.

Vector é uma forma de armazenar informações contíguas, onde os elementos podem ser acessados por iteradores (*iterators*) ou por *offsets* de ponteiros para os elementos. A diferença entre um *vector* e um *static array* é que o *vector* é alocado dinamicamente (a memória é alocada a medida que cresce), consequentemente ocupando mais memória enquanto o *static array* alocada memória estática. A tabela abaixo apresenta as suas operações e seus custos operacionais.

Operação	Custo	Operação	Custo
swap	constante	clear	linear(n)
operador=	linear(n)	assign	linear(n)
reserve	linear(n)	shrink_to_fit	linear(n)
erase	linear(n)	push_back	constante amortizada
emplace_back	constante amortizada	insert	linear
emplace	linear	resize	linear
pop_back	constant	at	constant
get_allocator	constant	operator[]	constant
front	constant	back	constant
empty	constant	size	constant
max_size	constant	capacity	constant
data	constant		

References