



11º Laboratório ECOP13A – Programação Orientada a Objetos
STL - parte 2 – 04 de julho 2025
Prof. André Bernardi (andrebernardi@unifei.edu.br)

Os exercícios da semana têm como objetivo principal fazer com que o aluno pratique o conceito de **estruturas de dados não-lineares em STL**. Siga as instruções:

1. **(ex01.cpp)** Utilizando a STL, escreva um programa em C++ para a demonstração do funcionamento de uma **Heap ou Fila de Prioridade**, incluída através do cabeçalho `<queue>`. Faça um programa que deve mostrar repetidamente um menu com as opções que podem ser escolhidas pelo usuário. Ele deve funcionar de maneira semelhante ao exemplo a seguir:

```
-----  
Programa de Heap STL  
-----  
  
1.Insira um elemento na heap  
2.Remova um elemento da heap  
3.Tamanho da heap  
4.Primeiro elemento da heap  
5.Sair  
Escolha (1-5): 1 (cin)  
Entre com o valor a ser inserido: 87 (cin)  
  
1.Insira um elemento na heap  
2.Remova um elemento da heap  
3.Tamanho da heap  
4.Primeiro elemento da heap  
5.Sair  
Escolha (1-5): 1 (cin)  
Entre com o valor a ser inserido: 92 (cin)  
  
1.Insira um elemento na heap  
2.Remova um elemento da heap  
3.Tamanho da heap  
4.Primeiro elemento da heap  
5.Sair  
Escolha (1-5): 1 (cin)  
Entre com o valor a ser inserido: 35 (cin)  
  
1.Insira um elemento na heap  
2.Remova um elemento da heap  
3.Tamanho da heap  
4.Primeiro elemento da heap  
5.Sair  
Escolha (1-5): 2 (cin)  
Elemento 92 removido do topo da heap  
  
1.Insira um elemento na heap  
2.Remova um elemento da heap  
3.Tamanho da heap  
4.Primeiro elemento da heap  
5.Sair  
Escolha (1-5): 3 (cin)  
Tamanho da heap: 2
```



```
1.Insira um elemento na heap
2.Remova um elemento da heap
3.Tamanho da heap
4.Primeiro elemento da heap
5.Sair
Escolha (1-5): 4 (cin)
Primeiro elemento da heap (topo): 87
```

```
1.Insira um elemento na heap
2.Remova um elemento da heap
3.Tamanho da heap
4.Primeiro elemento da heap
5.Sair
Escolha (1-5): 5(cin)
```

Programa finalizado!

2. (ex02.cpp) Utilizando a STL, escreva um programa em C++ para a demonstração do funcionamento de uma **árvore binária de busca balanceada** ou **set**, incluída através do cabeçalho `<set>`. Faça um programa que deve mostrar repetidamente um menu com as opções que podem ser escolhidas pelo usuário. A árvore poderá conter valores repetidos, portanto escolha a classe adequada (`multiset`). Ele deve funcionar de maneira semelhante ao exemplo a seguir:

```
=====
Implementação de Árvore (set) no STL
=====
1. Inserir elemento
2. Remover elementos com determinado valor
3. Exibir elementos em ordem
5. Exibir quantidade de elementos
6. Remover todos os elementos
7. Consultar quantidade de elementos com determinado valor
7. Sair

Escolha uma opção: (Usuário entra com cin)
```

Para mais informações sobre as funções membro do template list da STL, acesse o site com sua documentação em <http://www.cplusplus.com/reference/set/multiset/>. Este site pode ser muito útil. Ele contém a documentação de todas as classes e bibliotecas do C++.

Se tiver dúvidas entre em contato por e-mail! Não deixe para a última hora...

Este guia foi baseado em material desenvolvido pelo Prof. Carlos Henrique Valério de Moraes (UNIFEI).