

GREMLINS

tópicos abordados:

- Introdução;
- O projeto;
- A implementação;
- O projeto;
- Como compilar e executar;
- Testes realizados;
- Bugs;
- Observações;
- Sobre mim;

status do projeto: Em desenvolvimento

Introdução:

Este projeto implementa um gerenciador de memória em c++. Será utilizado por codigos escritos em c++ para fazer todo o gerenciamento de memória não deixando a cargo do programador criar ou deletar dados da memória alocada dinamicamente.

O projeto:

Esse projeto está sendo desenvolvido como uma atividade do semestre 2016.1 da turma de EDB1 e LP1 do professor Selan Rodrigues, um gerenciador de memória que maneja a memoria alocada dinâmica será

implementada em c++. O projeto está sendo implementado tomando como base os requerimento do que se pede no arquivo pdf enviado pelo professor.

A implementação:

Os seguintes arquivos são pertencentes a implementação da predição até o momento:

1. **src/main.hpp** : Contém o código da função main.
2. **src/drive_gremlins.hpp** : Contém o código das funções `simulacao()` e `StoragePoolTest()`.
3. **include/classStoragePool.hpp** : Contém a classe `StoragePool` que será utilizada para criar a classe `SLPool`.
4. **include/classSPool.hpp** : Contém a classe `SLPool`.
5. **include/classSPool.inl** : Contém a implementação dos métodos da classe `SLPool`.
6. **include/mempool_common.hpp** : Contém a implementação das sobrecargas de operadores `new`, `new[]` e `delete`.
7. **include/classEvents.hpp** : Contém a implementação da classe `Events`.
8. **Doxygen** : Arquivo de configuração para gerar a documentação do código com o doxygen.

Para maior entendimento do código escrito existe muitos comentários do que foi escrito, isso garantirá uma melhor compreensão do código por parte de quem vier a analisa-lo. O código foi implementado por Nalbert Gabriel Melo Leal, entretanto isso não significa que não existiu

a ajuda na lógica do código por parte de outros indivíduos. Na lógica das funções dos métodos `allocate()` e `free()` a ajuda de Matheus Alves foi importante, na implementação dos testes da função `simalacao()` a ajuda de Yuri Reinaldo e Adelino Afonso foi de extrema importância para compreensão do que deveria e como os testes deveriam ser feitos.

Como compilar e executar:

Para compilar o código é indicado o compilador `g++` pois esse foi o programa utilizado na compilação do código. Assumindo que os requerimentos/conselhos estejam sendo seguidos para compilar o código do vector usando o `g++` é necessário utilizar o seguinte comando no terminal dentro da pasta Vector:

```
g++ -std=c++11 -I include src/main.cpp -o bin/gremlins
```

Também pode se utilizar o `makefile`, basta digitar o comando a seguir que o código será compilado:

Para código otimizado (geralmente o gerenciador de memória ganha do SO):

```
make YES
```

Para código não otimizado (geralmente o SO ganha do gerenciador de memória):

```
make NO
```

Para compilar código para debug:

```
make DEBBUG
```

O arquivo gerado se encontra dentro da pasta bin do diretório do projeto e possui o nome gremlins. Para executar o arquivo basta entrar na pasta bin e digitar ./gremlins para ver um exemplo do uso do gerenciador de memória implementado.

Testes realizados:

Os testes realizados foram feitos com base no que pede o pdf enviado pelo professor Selan. Esses testes podem não refletir algum eventual bug que testes profundos no código revelariam. A ajuda de Yuri Reinaldo e Adelino Afonso foi extremamente importante para compreender como implementar os testes.

Bugs:

Os testes realizados não revelaram algum erro. Entretanto isso não significa que o programa está livre de erros.

Observações:

O código será postado no GitHub para possíveis avaliações futuras. O projeto é muito interessante e pode ter suas ideias fundamentais sendo utilizadas base para diversos projetos futuros.

Esse projeto se encontra no Bitbucket e Github nos links:

<https://bitbucket.org/nalbertg/gremlins>

e

<https://github.com/nalbertg/projeto-GREMLINS>

Sobre mim:

- **Aluno:** Nalbert Gabriel Melo Leal;
- **GitHub:** github.com/nalbertg ;
- **Bitbucket:** bitbucket.org/nalbertg/gremlins
- **email pessoal:** nalbertrn@yahoo.com.br;
- **email acadêmico/profissional:** nalbertg@outlook.com;