(../) ÿ **Rouille**

ESCALA PARA PROJETO CPP MÓDULO 09

(HTTPS://PROJECTS.INTRA.42.FR/PROJECTS/CPP- MODULE-09)

Você deve avaliar 1 aluno nesta equipe

Introdução

- Permaneça educado, cortês, respeitoso e construtivo durante
todo o processo de avaliação. O bem-estar da comunidade depende disso.

- Identificar junto à pessoa (ou ao grupo) avaliado as eventuais
disfunções do trabalho. Reserve um tempo para discutir
e debater os problemas que você identificou.

- Você deve considerar que pode haver alguma diferença na forma como seus
pares entenderam as instruções do projeto e o escopo de suas
funcionalidades. Sempre mantenha a mente aberta e avalie-o da forma
mais honesta possível. A pedagogia só é válida e somente se a avaliação
pelos pares for conduzida com seriedade.

Diretrizes

- Avaliar apenas os trabalhos que estão no repositório GiT
do aluno ou grupo.

- Verifique novamente se o repositório GiT pertence ao aluno ou ao
grupo. Certifique-se de que o trabalho seja para o projeto relevante
e verifique também se "git clone" é usado em uma pasta vazia.

- Verifique cuidadosamente se nenhum aliase malicioso foi usado para
enganá-lo e fazer com que você avalie algo diferente do conteúdo do
repositório oficial.

- Para evitar surpresas, verifique cuidadosamente se tanto os alunos
avaliadores quanto os avaliados revisaram os possíveis roteiros utilizados
para facilitar a avaliação.

- Caso o aluno avaliador ainda não tenha concluído aquele
determinado projeto, é obrigatória a leitura completa da
matéria antes de iniciar a defesa.

- Use os flags disponíveis nesta escala para sinalizar um repositório vazio,

programa não funcional, erro de norma, trapaça, etc. Nestes casos, a avaliação termina e a nota final é 0 (ou -42 em caso de trapaça). No entanto, com exceção da trapaça, você é incentivado a continuar discutindo seu trabalho (mesmo que não o tenha concluído) para identificar quaisquer problemas que possam ter causado essa falha e evitar repetir o mesmo erro no futuro.

- Lembre-se que durante o período da defesa não será permitida qualquer falta de segurança, nenhum outro encerramento inesperado, prematuro, descontrolado ou inesperado do programa, caso contrário a nota final será 0.
Utilize a bandeira apropriada.
Você nunca deverá editar nenhum arquivo, exceto o arquivo de configuração, se ele existir.
Se você quiser editar um arquivo, reserve um tempo para explicar os motivos ao aluno avaliado e certifique-se de que ambos concordam com isso.

- Você também deve verificar a ausência de vazamentos de memória. Qualquer memória alocada no heap deve ser liberada adequadamente antes do final da execução.
Você tem permissão para usar qualquer uma das diferentes ferramentas disponíveis no computador, como leaks, valgrind ou e_fence. Em caso de vazamento de memória, marque o sinalizador apropriado.

Anexos

input.csv (https://cdn.intra.42.fr/document/document/20226/input.csv)

assunto.pdf (https://cdn.intra.42.fr/pdf/pdf/101923/en.subject.pdf)

cpp_09.tgz (https://cdn.intra.42.fr/document/document/20227/cpp_09.tgz)

Testes preliminares

Se houver suspeita de trapaça, a avaliação termina aqui. Use o sinalizador "Cheat" para denunciá-lo. Tome esta decisão com calma, sabedoria e, por favor, use este botão com cautela.

Pré-requisitos

O código deve ser compilado em c++ e as flags -Wall -Wextra -Werror Não se esqueça que este projeto deve seguir o padrão C++98. Portanto, C++ 11 (e posteriores) NÃO são esperados.
O objetivo deste módulo é usar o STL. Então, o uso dos containers está autorizado.

Qualquer um destes significa que você não deve avaliar o exercício em questão:

- Uma função é implementada em um arquivo de cabeçalho (exceto funções de modelo).
- Um Makefile compila sem os sinalizadores necessários e/ou outro compilador que não seja c++.

Qualquer um destes significa que você deve sinalizar o projeto com "Proibido Função":

- Uso de uma função "C" (*alloc, *printf, free).
- Uso de função não permitida nas diretrizes do exercício.
- Uso de uma biblioteca externa ou recursos de versões diferentes do C++98.

Sim Não

Exercício 00: Troca de Bitcoin

Para este primeiro exercício, você deverá encontrar um makefile com as regras usuais de compilação e os arquivos solicitados no assunto.

Revisão de código

Verifique se um makefile está presente com as regras usuais de compilação.

Verifique no código se o programa usa pelo menos um contêiner.

O avaliado deve explicar por que optou por utilizar este container e não outro?

Caso contrário, a avaliação termina aqui.

Sim Não

Identificador de erro

Você deve ser capaz de usar um arquivo vazio ou um arquivo com erros (existe um exemplo básico no assunto). O programa não deve parar sua execução antes de ter realizado as operações em todo o arquivo passado como argumento.

Você pode usar uma data errada.

Você pode inserir um valor maior que 1000 ou menor que 0.

Se houver algum problema durante a execução a avaliação termina aqui.

Sim Não

Uso principal

Agora você deve usar o arquivo “input.csv” localizado no topo desta página.

Você pode modificar este arquivo com os valores desejados.

Você deve executar o programa com o arquivo input.csv como parâmetro.

Compare algumas datas manualmente com o valor especificado.

Caso a data não exista no banco de dados, o programa deverá utilizar a data inferior mais próxima.

Sim Não

Exercício 01: Notação Polonesa Reversa

Para este segundo exercício, você deverá encontrar um makefile com as regras usuais de compilação e os arquivos solicitados no assunto.

Revisão de código

Verifique se um makefile está presente com as regras usuais de compilação.

Verifique no código se o programa usa pelo menos um contêiner.

O avaliado deve explicar por que optou por utilizar este container e não outro?

Caso contrário, a avaliação termina aqui.

Se o container aqui escolhido estiver presente no primeiro exercício então a avaliação termina aqui.

Sim Não

Uso principal

Verifique se o programa funciona corretamente usando diferentes fórmulas de sua escolha.

O programa não é obrigado a lidar com expressões com parênteses ou números decimais.

Se houver algum problema durante a execução a avaliação termina aqui.

Sim Não

Uso avançado

Verifique se o programa funciona corretamente usando diferentes fórmulas de sua escolha.

Aqui estão alguns testes:

8 9 * 9 - 9 - 9 - 4 - 1 +
> Resultado: 42

9 8 * 4 * 4/2 + 9 - 8 - 8 - 1 - 6 -
> Resultado: 42

1 2 * 2/2 + 5 * 6 - 1 3 * - 4 5 * * 8 /
> Resultado: 15

Você pode usar os exemplos do tópico se não souber qual fórmula usar.

Se houver algum problema durante a execução a avaliação termina aqui.

Sim Não

Exercício 02: PmergeMe

Como sempre, deve haver testes suficientes para provar que o programa funciona conforme o esperado. Se não houver, não avalie este exercício. Se alguma classe sem interface não estiver na forma de classe canônica ortodoxa, não avalie este exercício.

Revisão de código

Verifique se um makefile está incluído nas regras usuais de compilação.

Verifique no código se o programa usa pelo menos dois contêineres.

Caso contrário, a avaliação termina aqui.

O avaliado deve explicar por que optou por utilizar esses recipientes e não outro?

Verifique no código se o algoritmo de classificação por inserção de mesclagem está presente
e é usado para cada contêiner. O algoritmo Ford-Johnson deve ser usado.
Espera-se uma breve explicação. Em caso de dúvida, a avaliação termina aqui.

Se um dos contentores aqui escolhidos estiver incluído num dos exercícios anteriores a
avaliação termina aqui.

Sim Não

Uso principal

Agora você pode verificar manualmente se o programa funciona corretamente usando entre 5 e 10 números inteiros positivos diferentes de sua escolha como argumentos do programa.

Se este primeiro teste funcionar e fornecer uma sequência ordenada de números, você poderá continuar.
Caso contrário, a avaliação termina agora.

Agora você deve verificar esta operação usando o seguinte comando como argumento para o programa:

Para Linux:

```
`shuf -i 1-1000 -n 3000 | tr "\n" " "
```

Para OSX:

```
`jot -r 3000 1 1000 | tr '\n' ' '`
```

Se o comando funcionar corretamente, o avaliado deverá saber explicar a diferença de tempo utilizado para cada contêiner selecionado.

Se houver algum problema durante a execução e/ou explicação a avaliação termina aqui.

Sim Não

Avaliações

Não se esqueça de verificar a bandeira correspondente à defesa

OK

Projeto excelente

Trabalho vazio

Trabalho incompleto

Compilação inválida

Trair

Colidir

Situação preocupante

Vazamentos

l Função proibida

Não é possível apoiar/explicar o código

Dê uma estrela a este repositório.

(https://github.com/rphlr/42-Evals)