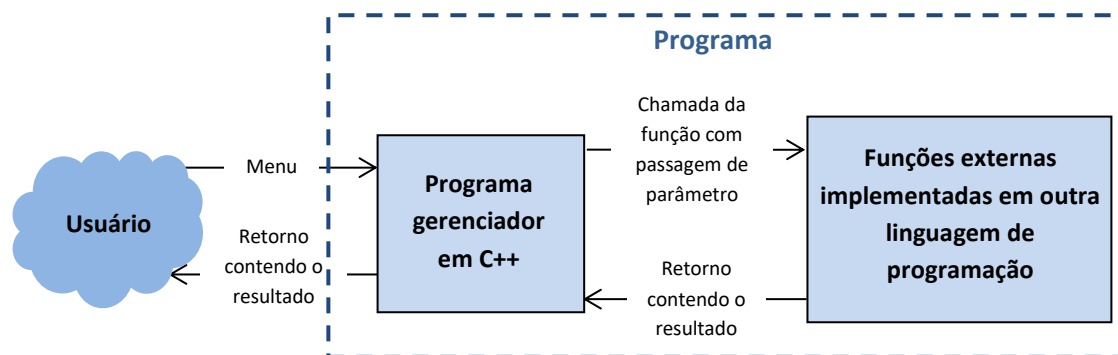


=== Trabalho ===

Este trabalho é dividido em duas etapas, cada uma com data de entrega e avaliação separadas. O objetivo final é criar um sistema que se sirva de uma interface com outra linguagem de programação para um propósito bem definido, como por exemplo, processar texto, armazenar dados, obter dados da Internet, utilizar funções mais bem consolidadas em outras linguagens de programação etc. Para isso, é preciso criar um sistema como o ilustrado abaixo:



A primeira etapa para a realização deste trabalho é identificar o seu objetivo. Uma vez definido o objetivo, esta etapa requer a apresentação do projeto do programa contendo as opções no menu oferecido aos usuários e as respostas esperadas. Ainda nessa etapa, o papel desempenhado pelo programa gerenciador em C++ e o papel desempenhado pelas funções externas devem ser igualmente apresentados. Dessa forma, as entradas e saídas da interface entre o gerenciador em C++ e as funções externas devem ser identificadas. Note que essas funções externas **tem que** pertencer a um programa ou script externo escrito em outra linguagem de programação (**importante: a outra linguagem não pode ser o C**). O mais importante é prever as interfaces no C++ que serão usadas para execução das funções externas. Um exemplo seria um sistema de detecção de spams, onde o programa gerenciador em C++ poderia gerenciar todos os e-mails do usuário, enquanto as funções externas poderiam classificar se o e-mail é um spam ou não, baseado em múltiplos critérios. Tais funções externas poderiam ser escritas em Perl e dessa forma haveria a necessidade de implementar a interface entre C++ e Perl no programa gerenciador. Outro exemplo seria a implementação de um sistema de gerenciamento de medicamentos em C++ que utilizaria dados armazenados em um banco MySQL. Dessa forma, o programa gerenciador em C++ teria que implementar as interfaces necessárias para execução de consulta e inserção/remoção de dados no banco. Possíveis dados necessários podem ser obtidos da Internet ou produzidos

sinteticamente. Nesse último exemplo, cabe ressaltar que as funções externas poderiam ser escritas em outra linguagem que permite maior facilidade de acesso ao banco de dados. Por exemplo, as funções externas podem ser em Python, que é usado para acessar um banco MySQL.

O resultado produzido na primeira etapa do trabalho, ETAPA 1, deve ser entregue ao Professor através da plataforma Google Classroom. A entrega será feita através de um relatório digital, de no máximo duas páginas, contendo apenas o objetivo do programa, a descrição das opções oferecidas no menu do usuário e das saídas do programa, o papel do programa gerenciador em C++ e o papel de cada uma das possíveis funções externas. Ainda, o formato das chamadas do programa em C++ às funções externas deve ser apresentado através da descrição de suas entradas e saídas. Uma dica é refazer o mesmo desenho deste enunciado de maneira adaptada ao trabalho proposto. O usuário deve ter no mínimo cinco (5) opções no menu, sendo que pelo menos uma delas deve se servir de resultados obtidos a partir da execução das funções externas. O prazo de entrega da ETAPA 1 é até o dia 03/06/24.

Após a definição do projeto, segue a etapa de desenvolvimento do programa. Esta etapa consiste na implementação do gerenciador em C++, na implementação das funções externas e, por fim, nas interfaces de integração entre eles. Todas as opções oferecidas no menu ao usuário (**ao menos 5 opções, como já mencionado**) devem ser programadas a partir de métodos ou funções em C++. É importante frisar que pelo menos uma dessas opções deve interagir com as funções externas. A interação é considerada somente se houver passagem de parâmetro para as funções externas e retorno do resultado correspondente ao programa gerenciador em C++, como visto na figura. **Não é permitido passar parâmetros ou retornar resultados através de arquivos e nem executar comandos através da função system.** O projeto definido na ETAPA 1 deve ser seguido, porém, algumas adaptações podem ser realizadas para o melhor funcionamento do programa, desde que descritas em relatório.

Note que o programa relativo ao gerenciador deve ser escrito em C++, utilizando ao máximo o paradigma de orientação a objetos. Por fim, as bibliotecas incluídas devem ser também as versões para C++, caso haja versão disponível para o C. O programa deve ser simples para o usuário e deve ser robusto ao mau uso. Por exemplo, uma operação não suportada ou uma entrada maliciosa não deve ser capaz de parar o programa ou retornar um valor inesperado. A interação do usuário com o programa deve ser limitada às operações possíveis oferecidas no menu.

Todos os códigos relativos à segunda etapa do trabalho, ETAPA 2, deverão ser entregues em formato digital (compactados em um único arquivo zip ou rar) contendo todos os arquivos necessários para a sua correta compilação e execução (arquivos

contendo os códigos fonte em C++ e na linguagem relativa às funções externas, arquivos de configuração e arquivos de entrada utilizados) através do Google Classroom. O código deve ser entregue em condição de ser compilado com g++ a partir de um arquivo Makefile incluído. O programa resultante deve ser executado corretamente. Além de todos os arquivos do programa, um relatório digital deverá ser entregue e o trabalho apresentado presencialmente em sala de aula. **O prazo de entrega da ETAPA 2 é até o dia 12/07/24.**

Todos os trabalhos deverão ser apresentados na data da entrega em até 10 minutos (os alunos serão convidados aleatoriamente). Os alunos deverão trazer slides para apresentação contendo explicações sobre o procedimento de compilação, sobre a utilização de todas as funções do programa e sobre o código referente ao uso das funções externas. Além dos slides, os alunos poderão usar vídeos que demonstrem os procedimentos apresentados.

Obs.1: O trabalho pode ser feito em dupla ou individualmente.

Obs.2: O relatório da ETAPA 1 deve seguir o formato descrito acima. Já o relatório da ETAPA 2 deverá conter uma introdução sobre o trabalho; uma descrição sobre os objetivos do programa; a implementação do programa contendo os critérios escolhidos, quais foram os motivos e trechos chave do código comentados e legíveis; casos de uso; e conclusão. Um modelo para elaboração de relatórios pode ser encontrado em:

<http://www.gta.ufrj.br/~miguel/docs/lingprog/modeloRelatorio.pdf>.

Obs.3: O código referente à ETAPA 2 deve ser compilável e executável em Linux.

Obs.4: Os trabalhos entregues fora do prazo terão pontos descontados proporcionalmente ao atraso, cabendo ao professor definir o peso dos descontos.

Obs.5: Trabalhos copiados, mesmo que parcialmente, receberão nota ZERO.
