

=== Trabalho ===

Este trabalho é dividido em três etapas, cada uma com data de entrega e avaliação separadas. O objetivo final é criar um sistema de processamento de texto capaz de obter dados não necessariamente formatados e extrair desses dados algumas métricas de interesse. Para isso, é preciso criar um sistema como o ilustrado abaixo:

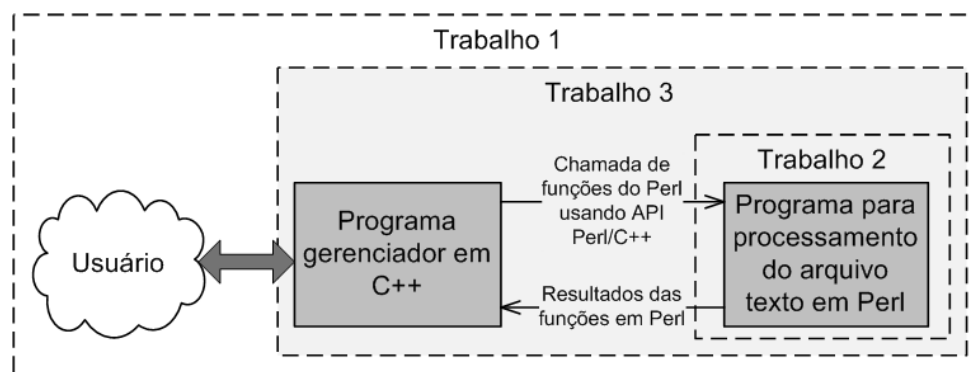


Figura 1: Arquitetura do sistema a ser implementado ao longo dos Trabalhos 1, 2 e 3.

A primeira etapa para a realização deste trabalho é identificar o problema a ser investigado. Uma vez definido o problema, as entradas oferecidas pelos usuários e as respostas esperadas do programa devem ser apresentadas. Ainda, na etapa de definição do problema, o papel desempenhado pelo programa gerenciador em C++ e o papel desempenhado pelo programa de processamento de arquivo texto em Perl no sistema devem ser igualmente apresentados. Dessa forma, as entradas e saídas da interface entre o gerenciador em C++ e o programa Perl devem ser identificadas. Um exemplo seria um sistema de detecção de spams, onde o programa gerenciador em C++ pode gerenciar todos os e-mails do usuário, enquanto o programa de processamento de arquivos apenas classifica o e-mail em spam ou não spam, baseado em múltiplos critérios. Note que a entrada do programa em Perl pode ser um arquivo texto gerenciado pelo programa em C++ e a saída pode ser uma classificação ou um conjunto de métricas a respeito do arquivo texto de e-mail.

A primeira etapa de definição do problema deve ser entregue ao Professor. **A entrega será feita através de um relatório impresso ou digital de uma página contendo apenas a definição do problema, as entradas do usuário e as saídas do programa, o papel do programa gerenciador em C++ (quais são as suas principais tarefas – cinco tarefas no mínimo) e o papel do programa de processamento de arquivo texto em Perl (quais são as suas principais tarefas – cinco tarefas no mínimo).** Ainda, as

entradas e saídas do programa em C++ e em Perl devem ser apresentadas no relatório. A entrega do Trabalho 1 deverá ser feita em sala de aula. O prazo de entrega do Trabalho 1 é até o dia 13/09/19.

Após a definição do problema, segue a etapa de desenvolvimento do programa para processamento de arquivo texto em Perl. Esse programa pode considerar como uma das entradas um arquivo texto a ser analisado e a saída é o resultado da análise ou o conjunto de métricas a respeito da análise realizada sobre o arquivo texto. Nessa etapa, o usuário interage diretamente com o programa em Perl, visto que o programa gerenciador em C++ ainda não foi implementado. A análise realizada pelo programa em Perl deve ser multicritério. Isso significa que o arquivo texto deverá ser analisado por múltiplos critérios (possivelmente um critério por função) com objetivos diferentes que, em conjunto, oferecem uma saída única ou um conjunto de saídas. Novamente, usando o exemplo do sistema de detecção de spams, a entrada do programa em Perl poderia ser um e-mail, ou nome de um arquivo, e a saída poderia ser um conjunto de notas para múltiplos critérios de avaliação ou uma nota única calculada a partir de uma composição das notas de cada critério. O exemplo abaixo ilustra um arquivo de e-mail e as possíveis saídas.

Arquivo de e-mail.

```
*****  
Date: Wed, 16 Jun 2010 10:38:03 -0300  
Date: Wed, 16 Jun 2010 10:38:03 -0300  
From: wdds2010 <wxyz2010@dominio.br>  
To: john@dominio.br  
Subject: WXYZ 2010 - Deadline: 20/10/2010 - FINAL
```

Desculpe-nos por eventuais duplicações

IV WXYZ 2010 - IV Workshop de XYZ
(workshop satélite do I Congresso Brasileiro de Qualquer Coisa)
Salvador, BA - 20 de dezembro de 2010

Viagra

<http://www.wxyz.br/>

Chamada de Artigos

=====

Deadline de Submissão (Prorrogado) Viagra: *** 20/10/2010 ***

Possíveis saídas.

Nota segundo critérios A, B, C, D, E = (5, 1, 4, 3, 10)

ou

Nota composta (p.ex. a soma) segundo critérios A, B, C, D, E = 23

Os trabalhos devem criar e utilizar funções definidas em bibliotecas próprias (arquivos *.pm) que devem ser possíveis de instalar **sem** permissão de super usuário. Não há necessidade de implementar todas as funções criadas em bibliotecas próprias. A **análise multicritério deverá ser baseada em, no mínimo, cinco análises diferentes semântica e sintaticamente**. Isso significa que, por mais que a ideia das análises seja diferente, essa diferença deve também refletir em código.

Os programas do Trabalho 2 deverão ser entregues em formato digital contendo todos os arquivos necessários para sua correta interpretação e execução (arquivos contendo os códigos fonte em Perl, módulos criados para o Perl, arquivos de configuração e arquivos de entrada utilizados). Além de todos os arquivos do programa, deverá ser entregue também um relatório digital ou impresso em sala de aula. As cinco análises realizadas devem ser implementadas e descritas no relatório claramente. O prazo de entrega do Trabalho 2 é até o dia 18/10/19.

A última etapa é desenvolver o programa gerenciador em C++ e integrá-lo com o programa em Perl desenvolvido no Trabalho 2. **O programa oferece pelo menos cinco opções ao usuário**. Essas opções são implementadas a partir de métodos ou funções em C++ programadas para o trabalho. É importante mencionar que pelo menos uma dessas opções deve interagir com o programa em Perl utilizando a API do Perl para C++. A interação é considerada somente se houver passagem de variável(is) para o programa em Perl e retorno de resultado(s) para o programa em C++. As funções desenvolvidas no Trabalho 2 devem continuar sendo usadas na parte do programa em Perl. No entanto, algumas adaptações podem ser realizadas para o melhor funcionamento do programa, desde que discriminadas em relatório. Utilizando mais uma vez como exemplo o sistema de detecção de spams, as cinco opções poderiam ser as seguintes, assumindo que o programa gerenciador tem acesso a todos os e-mails do usuário:

- Remoção de um e-mail do usuário;
- Exibição de todos os e-mails do usuário;
- Classificação de todos os e-mails (*método que utiliza o programa de processamento de texto em Perl – programa em C++ envia para o programa em Perl o nome do arquivo do e-mail a ser analisado e recebe de volta o resultado da avaliação*);
- Exibição do número total de e-mails;
- Exibição do número total de spams.

Note que o programa 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 C. O programa deve ser simples para o usuário e deve se proteger contra o 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.

Os programas do Trabalho 3 deverão ser entregues em formato digital contendo todos os arquivos necessários para a sua correta compilação e execução (arquivos contendo os códigos fonte em C++ e Perl, arquivos de cabeçalho *.h criados, módulos criados de Perl, arquivos de configuração e arquivos de entrada utilizados). O programa 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 ou impresso deverá ser entregue em sala de aula. O prazo de entrega do Trabalho 3 é até o dia 25/11/19.

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

Obs. 2: O relatório do Trabalho 1 deve seguir o formato descrito acima. Já os relatórios dos Trabalhos 2 e 3 deverão 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 o código comentado e legível; 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: Os programas dos Trabalhos 2 e 3 devem ser compiláveis e executáveis 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: Os alunos poderão ser escolhidos para explicar oralmente o trabalho realizado.

Obs. 6: Trabalhos copiados, mesmo que parcialmente, receberão nota ZERO.
