

EP1 - Raposa de Fogo

Motivação

Um novo navegador web está para ser lançado no mercado. Seu nome será *Raposa de Fogo*, em homenagem ao animal de estimação de Mr. Oliver, um dos idealizadores do projeto. Conta-se que Mr. Oliver, nas festas do mês de junho, acendeu uma fogueira em sua casa e acabou, sem querer, queimando o rabo de seu gato, que era considerado por todos como o gato mais esperto do bairro. Na ocasião, o pobre bichano ganhou o apelido de *Raposa de Fogo*. Indignados com os acontecimentos, os gatos do bairro onde Mr. Oliver mora passaram a arranhar e rasgar os sofás de suas casas, causando enorme incômodo em toda a vizinhança. Na tentativa de reverter a situação, Mr. Oliver decidiu destinar parte da renda que for arrecadada com a venda do *Raposa de Fogo* (o navegador, não o gato, é claro!) para a compra de brinquedos e ração “de primeira” para os fiéis amigos do *Raposa de Fogo* (agora sim, o gato!). Você, que certamente adora gatos, está sendo convocado a auxiliar no desenvolvimento deste projeto.

Sua Tarefa

Sua tarefa será a seguinte. Seu programa deverá ler um arquivo texto contendo um inteiro n , indicando o número de comandos a serem lidos. Em cada uma das próximas n linhas, haverá um comando a ser interpretado por seu programa. Os comandos possíveis são os seguintes: `visit`, `back` e `download`. Antes de cada comando, haverá o parâmetro `<instante>`, que será sempre um inteiro positivo, maior que o instante anterior, indicando em qual instante o respectivo comando foi executado pelo usuário. Abaixo, está descrito o formato em que os comandos serão digitados:

- `<instante> visit <url> <tamanho>`: este comando indica que o usuário deseja carregar a página cujo link está informado no parâmetro `<url>`. O parâmetro `<tamanho>` será sempre um inteiro positivo, indicando o tamanho, em megabytes, da página a ser carregada.
- `<instante> download <url> <tamanho>`: este comando indica que o usuário deseja baixar o arquivo cujo link está informado no parâmetro

<url>. O parâmetro <tamanho> será sempre um inteiro positivo, indicando o tamanho, em megabytes, do arquivo a ser baixado.

- <instante> back: este comando indica que o usuário deseja interromper o carregamento da página atual (caso alguma página esteja sendo carregada no respectivo instante) e voltar a carregar a página que estava sendo carregada anteriormente. Como, nesta versão do *Raposa de Fogo*, não haverá comando *forward*, o conteúdo carregado da página atual pode ser “jogado fora”.

Seu programa deve funcionar da seguinte forma. A cada instante, seu programa deve primeiro processar o comando referente ao instante atual (veja a Tabela 1), caso haja algum, e depois realizar alguma das tarefas descritas na Tabela 2.

Tabela 1: Processamento dos Comandos

Comando	Ação
visit A	Coloca a página A na pilha de páginas.
download A	Coloca o arquivo A na fila de arquivos.
back	Caso a pilha não esteja vazia, remove a página que está no topo da pilha. Caso a pilha esteja vazia ou fique vazia após a remoção do elemento que estava no topo, imprimir a seguinte mensagem: <code>page not found</code> .

Tabela 2: Processamento das Tarefas

Situação	Ação
Página que está no topo da pilha de páginas ainda não foi totalmente carregada.	<p>Carregar 1 megabyte da respectiva página (digamos, página A) e depois, caso a página tenha sido totalmente carregada, imprimir a seguinte mensagem ao usuário:</p> <p><code><instante> web page A showed to user.</code></p> <p>Caso contrário, imprimir a seguinte mensagem ao usuário:</p> <p><code><instante> loading web page A n/m.</code></p> <p>onde <i>n</i> e <i>m</i> correspondem ao número de megabytes já carregados e ao número total de megabytes para carregar a página, respectivamente.</p>
Pilha de páginas está vazia ou página que está no topo da pilha de páginas já foi totalmente carregada em algum instante anterior.	<p>Caso a fila de arquivos a serem baixados esteja vazia, não há nada a fazer. Caso contrário, retirar um arquivo da fila (digamos, arquivo A) e carregar 1 megabyte do respectivo arquivo. Caso o arquivo tenha sido totalmente baixado, imprimir a seguinte mensagem ao usuário:</p> <p><code><instante> file A download complete successfully.</code></p> <p>Caso contrário, inserir o arquivo A de volta na fila e imprimir a seguinte mensagem ao usuário:</p> <p><code><instante> loading file A n/m.</code></p> <p>onde <i>n</i> e <i>m</i> correspondem ao número de megabytes já baixados e ao número total de megabytes do arquivo, respectivamente.</p>

Comando <code>back</code> foi processado no instante atual e a página que agora está no topo da pilha (digamos, página A) já foi totalmete carregada.	<p>Imprimir a seguinte mensagem:</p> <p><instante> web page A showed to user.</p> <p>Em seguida, processar a próxima tarefa, de acordo com as regras descritas nas linhas acima (neste caso, pode acontecer de duas mensagens serem impressas no mesmo instante).</p>
---	---

Um exemplo

A seguir, é mostrado um exemplo de entrada e a respectiva saída.

Conteúdo do arquivo de entrada

```

14
1 visit page1 2
12 download file1 4
13 visit page2 4
14 download file2 3
15 visit page3 3
16 back
17 download file3 6
18 download file4 4
19 visit page4 3
20 visit page5 3
33 back
34 back
35 back
36 back

```

Conteúdo do arquivo de saída

```

1 loading web page page1 1/2.
2 web page page1 showed to user.
12 loading file file1 1/4.
13 loading web page page2 1/4.
14 loading web page page2 2/4.
15 loading web page page3 1/3.
16 loading web page page2 3/4.
17 web page page2 showed to user.
18 loading file file1 2/4.
19 loading web page page4 1/3.
20 loading web page page5 1/3.
21 loading web page page5 2/3.
22 web page page5 showed to user.
23 loading file file2 1/3.
24 loading file file3 1/6.
25 loading file file4 1/4.
26 loading file file1 3/4.
27 loading file file2 2/3.
28 loading file file3 2/6.
29 loading file file4 2/4.
30 file file1 download complete successfully.
31 file file2 download complete successfully.
32 loading file file3 3/6.
33 loading web page page4 2/3.
34 web page page2 showed to user.
34 loading file file4 3/4.
35 web page page1 showed to user.
35 loading file file3 4/6.
36 page not found.
36 file file4 download complete successfully.
37 loading file file3 5/6.
38 file file3 download complete successfully.

```

Instruções adicionais

Submeta seu EP até às 23:30 do dia 17 de setembro através do site edisciplinas.usp.br (o sistema aceitará submissões até às 23:55, mas caso você tente submeter em um horário muito próximo ao encerramento e tiver problemas, fica por sua conta e risco...). O arquivo submetido deve estar no formato zip e deve conter apenas os códigos-fonte (somente os arquivos .cpp e .h, não adicione os executáveis).

Não será permitido utilizar nenhuma API que implemente as estruturas de dados necessárias (pilha e fila). Faça a sua própria implementação de pilha e fila. Caso haja casos de plágio, todos os envolvidos serão convocados a prestar contas e, dependendo da gravidade do caso, podem ser sumariamente reprovados na disciplina.

Bom Trabalho!