

Exercício para Nota

23/09/2023 – período tarde



Ao entregar esta atividade no Teams, será garantida a presença para o dia 23/09/2023 no período da tarde.

Histórico de Navegação

Ao navegar na internet, a cada site acessado, o endereço é armazenado em uma pilha. Ao clicar em voltar, pode-se acessar novamente o endereço anteriormente visitado.

Faça um programa que simule as funcionalidades de acessar um site e voltar. Ao empilhar, armazenar o endereço http. Ao desempilhar, retornar o endereço http. Não há limite de sites que podem ser acessados. Armazene o endereço http. A entrada de dados deve ocorrer dentro da função de empilhamento.

Os protótipos das funções estão definidas.

As estruturas de dados estão definidas como o modelo apresentado:

```
struct URL {
    string url;
    int dia;
    int mes;
    int ano;
};

struct Node {
    URL url;
    Node *next;
};

struct Pilha {
    Node *topo;
    int tamanho;
};
```

- 1) Desenvolver a função inicializar. (**void inicializar(Pilha *);**). Esta função deve inicializar o tamanho da pilha como 0 e apontar o topo da pilha para NULL. **(1 ponto)**
- 2) Desenvolver a função navegar (**void navegar(Pilha *, string);**). Esta função recebe a URL acessada no segundo argumento. A função deve descobrir o dia, mês e ano corrente e armazenar na estrutura URL. Após, deve empilhar URL na pilha e incrementar o tamanho. **(1 ponto)**
- 3) Desenvolver a função navegar (**void navegar(Pilha *, string, int, int, int);**). Esta função recebe a URL acessada no segundo argumento, o dia no terceiro argumento, o mês no quarto argumento e o ano no quinto argumento. A função deve descobrir o dia, mês e ano corrente e armazenar na estrutura URL. Após, deve empilhar URL na pilha e incrementar o tamanho. **(1 ponto)**
- 4) Desenvolver a função voltar (**URL voltar(Pilha *);**). Esta função retorna a estrutura URL no topo da pilha. **Cuide para que o programa não gere leaks de memória. (2 pontos)**
- 5) Desenvolver a função imprime_historico (**void imprime_historico(Pilha *);**). Esta recebe um ponteiro da estrutura Pilha, imprime o total de URLs empilhadas e em seguida imprime todas as URLs empilhadas. **(2 ponto)**

- 6) Desenvolver a função `imprime_url (void imprime_url(URL);)`. Esta recebe a estrutura URL e imprime todos os campos da estrutura (URL, dia, mês, ano). **(2 ponto)**
- 7) Desenvolver a função `destruir_historico (void destruir_historico(Pilha *);)`. Esta recebe a estrutura Pilha e desempilha todos os elementos que foram empilhados dinamicamente. **(1 ponto)**
- 8) Descomentar os comentários entre as linhas 45 e 64. Ao descomentar, compilar e executar, espera-se que a saída do programa seja semelhante ao apresentado na próxima página. **(0 pontos)**

Saída esperada ao descomentar entre as linhas 45 e 64 (função main)

Quantidade de URLs: 8 no histórico.

=====

<http://www.gmail.com>
<http://www.fatecararas.sp.gov.br>
<http://www.facebook.com>
<http://www.google.com>
<http://www.fatecararas.sp.gov.br>
<http://www.globo.com>
<http://www.facebook.com>
<http://www.google.com>

20/9/2023	http://www.gmail.com
20/9/2023	http://www.fatecararas.sp.gov.br
20/9/2023	http://www.facebook.com

Quantidade de URLs: 5 no histórico.

=====

<http://www.google.com>
<http://www.fatecararas.sp.gov.br>
<http://www.globo.com>
<http://www.facebook.com>
<http://www.google.com>

Desempilhado a URL: <http://www.google.com>
Desempilhado a URL: <http://www.fatecararas.sp.gov.br>
Desempilhado a URL: <http://www.globo.com>
Desempilhado a URL: <http://www.facebook.com>
Desempilhado a URL: <http://www.google.com>