

Disciplina: EDA

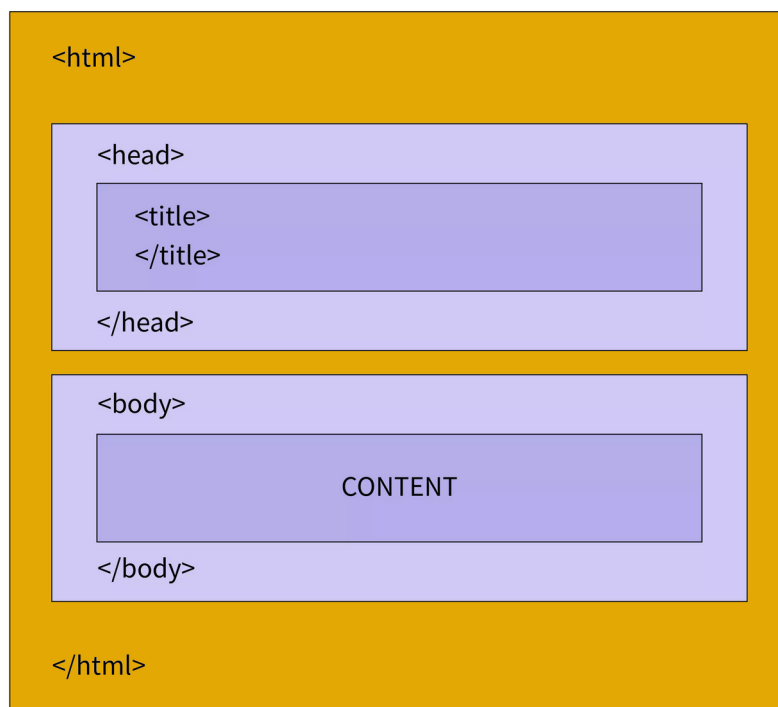
Entrega 26/11/2013

Trabalho de implementação sobre Pilhas

1. Contexto

O trabalho consiste na implementação de um programa verificador de estrutura *HTML* simples utilizando a estrutura de dados pilha.

O programa deve ser capaz de verificar se todas as *tags* abertas dentro do bloco laranja na figura abaixo, entre `<html>` e `</html>`, estão abertas e fechadas numa ordem válida.



Exemplo de arquivo HTML válido:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>

  <body>
    The content of the document.....
  </body>
</html>
```

Exemplo de arquivo HTML inválido:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  <body>
</head>
  The content of the document.....
  </body>
</html>
```

No exemplo acima a *tag* mais externa “head” é fechada antes da sua *tag* aninhada “body”, isto caracteriza um erro.

Em um arquivo *HTML*, *tags* são abertas segundo o padrão *menor-que*, *nome-da-tag* e *maior-que* (<nome_da_tag>) e fechadas segundo o padrão *menor-que*, *barra*, *nome-da-tag* e *maior-que* (</nome_da_tag>).

Existem algumas *tags* que não necessitam do fechamento, é o caso da *tag* <!DOCTYPE> utilizada nos exemplos anteriores.

2. Requisitos do programa:

- A implementação do programa deve ser feita em C, e a Pilha utilizada deve ser alguma das implementações discutidas em sala;
- A informação do arquivo a ser verificado pelo programa deve ser passado como parâmetro da linha de comando. (ex. ./verificador-html arquivo.html)
Dica: utilize *argc* e *argv*;
- O programa também deve emitir mensagem de erro a cada incorreção:
Exemplo de mensagem:
 - “ERRO: na linha xxx esperado <\body>, recebido <\head>”
 - “ERRO: a linha tttt encontrada tag zzz de fechamento sem prévia tag abertura”
- O programa deve gerar um arquivo de log contendo todas as mensagens de erro exibidas durante a execução;

- O nome da *tag* é a única parte utilizada na comparação entre *tags*. O nome da *tag* é a primeira *string* contida na *tag* até o primeiro espaço:

No quadro abaixo, as *tags* são: label; input; datalist e option;

```
<label for="fruits">Choose a fruit:</label>
<input list="fruitList" id="fruits" name="fruits">

<datalist id="fruitList">
  <option value="Apple">
  <option value="Banana">
  <option value="Cherry">
  <option value="Date">
  <option value="Elderberry">
  <option value="Fig">
</datalist>
```

- O nome de uma *tag* é formada de no máximo cinquenta (50) caracteres.
- A *tags* que não necessitam fechamento são apresentadas na tabela abaixo:

Tags que não necessitam fechamento
<!DOCTYPE> <input> <frame>

- Não é necessário tratar a *tag* de comentário <!-- ... -->

3. Formato de entrega e avaliação do trabalho:

Subir um zip identificado com os nomes dos autores, os conteúdos devem estar distribuídos em pastas contendo:

1. Todos os códigos fontes;
2. Todos os arquivos de entrada e/ou saída utilizados;
3. relatório informando os nomes dos autores e a arquitetura do sistema;

Avaliação será baseada em:

1. Funcionamento de acordo com o especificado;
2. apresentação da árvore na forma de uma estrutura de dados abstratos (TDA);
3. Compilação sem erros ou warnings;
4. Formato de entrega;
5. Apresentação do trabalho (entrevista).