Universidade Federal do Espírito Santo – Departamento de Informática

Estruturas de Dados (INF092092) - Período: 2017/2 Segunda Prova Parcial - 21/12/2017

Prof^a Patrícia Dockhorn Costa Email: pdcosta@inf.ufes.br

Regras importantes: A prova é sem consulta. Em caso de cola, a prova vale zero. A clareza e organização são importantes critérios de avaliação. Todo código deve ser claramente comentado.

Considere um programa que tem por objetivo contabilizar as presenças e faltas de alunos de uma turma da UFES. O programa recebe como entrada: (a) um arquivo de configuração com o número N de aulas lecionadas no semestre e os nomes dos alunos matriculados; (b) N arquivos contendo a presença dos alunos da turma em cada aula (onde N é o número de aulas lecionadas).

O quadro a seguir mostra um exemplo de um arquivo de configuração de entrada:

N 5 BRUNO DENER DENIS FABIO FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	BRUNO DENER DENIS FABIO FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	AT . F.	
DENER DENIS FABIO FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	DENER DENIS FABIO FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO		
DENIS FABIO FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	DENIS FABIO FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	BRUNO	
FABIO FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	FABIO FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	DENER	
FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	FELLIPE FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	DENIS	
FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	FERNANDO FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	FABIO	
FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	FILIPE FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	FELLIPE	
FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	FLAVIO GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	FERNANDO	
GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	GABRIEL IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	FILIPE	
IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	IGOR JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	FLAVIO	
JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	JOAO LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	GABRIEL	
LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO	LEONARDO LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	IGOR	
LUCAS LUIZ MATHEUS OTAVIO PABLO	LUCAS LUIZ MATHEUS OTAVIO PABLO PEDRO	JOAO	
LUIZ MATHEUS OTAVIO PABLO	LUIZ MATHEUS OTAVIO PABLO PEDRO	LEONARDO	
MATHEUS OTAVIO PABLO	MATHEUS OTAVIO PABLO PEDRO	LUCAS	
OTAVIO PABLO	OTAVIO PABLO PEDRO	LUIZ	
PABLO	PABLO PEDRO	MATHEUS	
	PEDRO	OTAVIO	
PEDRO		PABLO	
	TASSIO	PEDRO	
TASSIO		TASSIO	

O número N na primeira linha representa o número de aulas (que corresponde ao número de arquivos contendo as chamadas) e as linhas consecutivas contém os nomes dos alunos da turma.

O quadro a seguir mostra um exemplo de um arquivo de presença (P para presença e F para falta):

BRUNO P
DENER P
DENIS P
FABIO F
FELLIPE F
FERNANDO F
FILIPE P
FLAVIO P
GABRIEL P
IGOR F
JOAO F
LEONARDO F
LUCAS P
LUIZ F
MATHEUS P
OTAVIO F

```
PABLO F
PEDRO P
TASSIO P
```

Escreva um programa que leia as informações de um arquivo de configuração chamado "entrada.txt", e leia os outros arquivos com as presenças dos alunos em cada aula. Os arquivos serão nomeados de acordo com a sequência das aulas. Se N for 5, por exemplo, serão considerados 5 arquivos de chamada: entrada1.txt, entrada2.txt, entrada3.txt, entrada4.txt e entrada5.txt.

O arquivo de saída (saida.txt) deve fornecer as informações contabilizadas sobre faltas e presenças:

```
BRUNO 3P 2F
DENER 5P OF
DENIS 4P 1F
FABIO 1P 4F
FELLIPE 2P 3F
FERNANDO 3P 2F
FILIPE 5P OF
FLAVIO 1P 4F
GABRIEL 5P OF
IGOR 4P 1F
JOAO 3P 2F
LEONARDO 1P 4F
LUCAS 5P OF
LUIZ 4P 1F
MATHEUS 3P 2F
OTAVIO 4P 1F
PABLO OP 5F
PEDRO 5P OF
TASSIO 2P 3F
```

Regras importantes:

- 1) Deve-se usar uma **Tabela Hash** como estrutura de dados para armazenamento e busca de dados. O tratamento de colisão deve ser feito com **listas encadeadas**;
- 2) O nome do aluno deve ser considerado como chave de busca (ou seja, deve ser usado como entrada para a função de hash);
- Somente nomes simples serão utilizados;
- 4) Utilize tipos de dados estruturados;
- 5) As turmas da UFES têm no máximo 100 alunos;
- 6) NÃO utilize variáveis globais;
- 7) Libere toda a memória alocada;
- 8) O arquivo com a função main deve ter o seu nome: <nome>.c.

Dicas:

Funções de manipulação de strings < string.h>:

- strlen: Retorna o tamanho da string (sem o '\0').
- strcpy: Faz a cópia de strings, desde que a string destino tenha tamanho suficiente. strcpy (destino, origem).
- strcmp: Compara duas strings. strcmp (str1, str2). Retorna:
 - o 0 se str1 for igual a str2.
 - <0 se str1 for menor que str2 (como a sequência das palavras no dicionário)
 - > >0 se str1 for maior que str2

Para compilar e gerar o executável

```
gcc -c arq1.c (compila o arquivo e gera arquivo arq1.o)
gcc -c arq2.c (compila o arquivo e gera arquivo arq2.o)
gcc arq1.o arq2.o(faz a linkagem e gera o executável a.out)
```

Sugestões para leitura/escrita de arquivo

fopen (para abrir o arquivo)

```
FILE* fp;
fp = fopen ("entrada.txt", "rt");
if (fp == NULL) {
    printf ("Erro na abertura do arquivo!\n");
    exit(1);
}
```

fclose (para fechar o arquivo)

```
int fclose (FILE* fp);
```

• fgetc (para ler caracter a caracter)

```
//exemplo de função que lê uma palavra
int le palavra( FILE *fp, char *s ) {
  int \overline{i} = 0;
  int c;
  while ((c = fgetc(fp))) != EOF) {
    if (isalpha(c))
      break;
  if (c == EOF)
    return 0;
  else
    s[i++] = c;
  while (i < NPAL-1 \&\& (c = fqetc(fp))! = EOF \&\& isalpha(c))
   s[i++] = c;
  s[i] = ' \setminus 0';
  return 1;
}
```

fprintf (para escrever no arquivo)

```
fprintf(arquivo, %s, string)
fprintf(arquivo, %d, inteiro)
fprintf(arquivo, %.2f, float) //imprime 2 casas decimais
```