

Instituto de Computação da UNICAMP

Disciplina MC102: Algoritmos e Programação de Computadores - Turmas EF

Laboratório Nº 12. Peso 1.

Prazo de entrega: **15/05/2015 às 23:59:59**

PROFESSOR: Alexandre Xavier Falcão

MONITORES: João do Monte Gomes Duarte

MONITORES: Jadisha Yarif Ramírez Cornejo

MONITORES: Takeo Akabane

MONITORES: Eduardo Spagnol Rossi

MONITORES: Guilherme Augusto Sakai Yoshike

Tabela

O problema de busca de ocorrências de uma determinada string em um texto é de fundamental importância em diversas aplicações. Neste laboratório você deve fazer um programa que faça a busca de uma palavra em uma tabela que representa um cadastro de alunos. O seu programa deve explicitar cada ocorrência da palavra nesta tabela.

Será dada uma tabela com 5 colunas definidas como: "Ra", "Nome", "Curso", "Telefone" e "Estado". Os itens que ocorrem em cada coluna correspondem a uma string e possuem um tamanho específico: a coluna "RA" possui itens com exatamente 6 caracteres, a coluna "Nome" tem itens com, no máximo, 100 caracteres, a coluna "Curso" tem itens com exatamente 3 caracteres, a coluna "Telefone" tem itens com exatamente 14 caracteres, e finalmente a coluna "Estado" tem itens com exatamente 2 caracteres.

Dada uma tabela, onde cada linha representa os dados de um aluno específico, e dada uma palavra de busca str, o seu programa deve verificar se esta palavra é uma substring de algum elemento da tabela (exceto a primeira linha que corresponde aos nomes das colunas). Caso esta palavra seja uma substring de algum elemento desta tabela, o seu programa deve indicar em que posições esta palavra ocorre, indicando o nome da coluna e o número da linha. Caso esta palavra não ocorra na tabela, o seu programa não deve escrever nada. **Caso a palavra ocorra mais de uma vez na mesma linha e coluna, o seu programa só deve indicar a linha e coluna uma vez.**

A entrada será dada pela entrada padrão no seguinte formato: a primeira linha é a palavra a ser encontrada na tabela, a segunda linha é um número inteiro representando a quantidade de linhas N da tabela, que é no máximo 1000, em seguida é dado a tabela no formato [CSV](#). Isto significa que é dado uma linha por vez da tabela, onde os valores das colunas são separados por vírgula. Será dada uma linha inicial com os nomes das colunas e em seguida N linhas contendo os dados da tabela. Para este laboratório você pode supor que nenhum elemento da tabela terá vírgula nem aspas.

Por exemplo, a entrada

424

5

RA, Nome, Curso, Telefone, Estado

101010, Vinicius de Moraes, 424, (21)99898-0107, RJ

124246, Adoniram Barbosa, 424, (11)98989-0424, SP

087654, Murilo Rubiao, 321, (31)99632-5874, MG

056789, Atonio Carlos jobim, 125, (22)42424-1111, RJ

112233, Mario de Andrade, 424, (19)98426-3571, SP

Representa a busca da string "424" na tabela abaixo

RA	Nome	Curso	Telefone	Estado
101010	Vinicius de Moraes	424	(21)99898-0107	RJ
124246	Adoniram Barbosa	424	(11)98989-0424	SP
087654	Murilo Rubiao	321	(31)99632-5874	MG
056789	Atonio Carlos jobim	125	(22)42424-1111	RJ
112233	Mario de Andrade	424	(19)98426-3571	SP

O programa deve exibir todos os elementos da tabela em que ocorre esta palavra. Então, no exemplo dado, o programa deveria imprimir

Linha 1, Coluna "Curso"

Linha 2, Coluna "RA"

Linha 2, Coluna "Curso"

Linha 4, Coluna "Telefone"

Linha 5, Coluna "Curso"

Implementação

Você deve fazer um programa que verifica se uma palavra `str` é uma substring de elementos em uma tabela de strings. Caso seja uma substring de algum elemento, o programa deve listar todas as ocorrências na tabela na seguinte ordem: da primeira para a última linha e da primeira para a última coluna. Para cada elemento da tabela tal que `str` é uma substring, o programa deve imprimir

Linha `i`, Coluna "Nome da Colunas"

onde `i` é o número da linha e "Nome da Coluna" é o nome da coluna.

A tabela terá, no máximo, 1000 linhas. O tamanho máximo da substring a ser buscada terá tamanho máximo de 100 caracteres.

A única biblioteca que poderá ser usada é a `stdio.h`.

Entrada

A primeira linha é a palavra que deve ser buscada. A segunda linha é um inteiro `N` indicando o

número de linhas que a tabela terá. As linhas seguintes são formadas pela tabela no formato CSV, uma linha da tabela em cada linha de texto. Uma linha inicial sempre terá os nomes das 5 colunas denominadas "RA", "Nome", "Curso", "Telefone" e "Estado". As próximas N linhas conterão os elementos da tabela, uma linha por vez. Os elementos da coluna "RA" sempre terão 6 caracteres, da coluna "Nome" terão, no máximo, 100 caracteres, da coluna "Curso" terão exatamente 3 caracteres, da coluna "Telefone" terão exatamente 14 caracteres e da coluna "Estado" terão exatamente 2 caracteres.

Saída

A saída é formada por M linhas, sendo M o número de elementos da tabela em que ocorre a palavra buscada. Para cada ocorrência deverá ser impresso:

Linha i, Coluna "Nome da Coluna"

onde i é o número da linha e "Nome da Coluna" é o nome da coluna em que ocorre a palavra. (OBS: Deve haver exatamente uma quebra de linha após a última linha impressa).

Importante: para ler cada linha da tabela use o scanf com a opção %[^\n]. Para isto você deve usar o comando scanf("%[^\n]s", str); seguido de getchar(); O comando scanf("%[^\n]s", str); lerá uma string até o caracter '\n', porém não lerá este caracter. O comando getchar(); irá pegar o próximo comando, o '\n'. Deste modo a continuação do programa poderá ler as próximas strings.

Exemplos de execução

Exemplo 1

424

5

RA, Nome, Curso, Telefone, Estado

101010, Vinicius de Moraes, 424, (21)99898-0107, RJ

124246, Adoniram Barbosa, 424, (11)98989-0424, SP

087654, Murilo Rubiao, 321, (31)99632-5874, MG

056789, Antonio Carlos jobim, 125, (22)42424-1111, RJ

112233, Mario de Andrade, 424, (19)98426-3571, SP

Linha 1, Coluna "Curso"

Linha 2, Coluna "RA"

Linha 2, Coluna "Curso"

Linha 4, Coluna "Telefone"

Linha 5, Coluna "Curso"

Nota: Textos em azul denotam dados de entrada, isto é, a serem lidos pelo programa.

Textos em vermelho denotam dados de saída, ou seja, a serem impressos pelo programa.

Observações

- O número máximo de submissões é 15;
- O seu programa deve estar completamente contido em um único arquivo denominado `tabela.c`;
- Para a realização dos testes do SuSy, a compilação se dará da seguinte forma: `gcc -std=c99 -pedantic -Wall -lm -o tabela tabela.c`;
- Você deve incluir, no início do seu programa, uma breve descrição dos objetivos do programa, da entrada e da saída, além do seu nome e do seu RA.

Critérios importantes

Independentemente dos resultados dos testes do SuSy, o não cumprimento dos critérios abaixo implicará nota zero nesta tarefa de laboratório.

- O único header aceito para inclusão é `stdio.h`.

Voltar