

**Universidade Federal de Pernambuco**  
**Cin -Centro de Informática**  
**Lista 4 – Introdução a Programação**

**Prof: Adriano Sarmiento**

**Data: 20.04.2011**

**Data de entrega: 07.05.2011**

**Considerações:**

- **É proibido usar a biblioteca conio.h;**
- **Leia a lista toda o quanto antes, para evitar más interpretações e muitas dúvidas em cima da hora;**
- **Envie uma prévia da lista, pelo menos um dia antes da data final da entrega, para o caso de acontecer algum imprevisto;**
- **A lista é para ser feita individualmente. Qualquer tentativa de cópia acarretará o zeramento da lista de todos os envolvidos;**
- **Em caso de dúvida, envie email para [listaip@googlegroups.com](mailto:listaip@googlegroups.com).**
- **Atenção para a liberação da memória no final dos programas, será cobrado que o espaço alocado no decorrer do programa seja totalmente liberado no final do mesmo.**

1)Faça uma função que receba uma string, e quatro parâmetros passados por referência, dois serão o endereço de um caractere, onde no final da função, um receberá o caractere que mais aparece na string, e outro que receberá o que menos aparece na string. E os outros dois parâmetros serão o endereço de um inteiro, onde um receberá a quantidade de vezes que o maior aparece na string, e outro que receberá a quantidade de vezes que o menor aparece na string. A função deverá ser do tipo void. Os caracteres que devem ser considerados serão somente os alfa numéricos (a-z, 0-9). Maiúsculas e minúsculas deverão ser tratadas igualmente ('A' é igual a 'a'). Em caso de empate, a letra escolhida deve ser àquela com o menor valor ASCII (quando as letras em questão estão todas em maiúsculas ou todas em minúsculas – EX : entre 'E' e 'c', a letra 'c' deve ser escolhida, pois 'c' < 'e' na tabela ASCII).

Faça um programa que receba uma string do usuário e utilize esta função. Imprimindo o valor das quatro variáveis passadas por referência.

Ex:

Entrada:

Eu amo programacao

Saida

Caracter mais frequente : a

Frequência : 4

Caracter menos frequente : c

Frequência : 1

2) Existe um terreno que é uma matriz de char 7x7, nesse terreno existem 2 seres, um Homem e um Cachorro. Eles sempre começam nas posições abaixo (deve-se respeitar o limite do terreno)

```
H-----  
-----  
-----  
-----  
-----  
-----  
-----C
```

O cachorro pode se mover 1 casa (mesmo que tenha o homem a sua frente, seu movimento é randômico, nunca se sabe onde ele poderá estar). A cada rodada deve-se imprimir o terreno com as novas posições dos elementos na tela. E o cachorro nunca ficará parado até que seja pego (e quando for pego não se deve imprimir a configuração final do terreno). O movimento do homem você dará a cada iteração do laço até que as posições do homem e do cachorro sejam a mesma, então uma mensagem deve ser mostrada "Você pegou o cachorro". Quando a ação é dada então o cachorro se movimentará. O homem também pode se mover somente por 1 casa.

Você como aluno de Engenharia da Computação já viu operações com ponteiros e deverá aplicar nessa questão. E tudo que se pode fazer em relação a ponteiros que você conhece .

Os movimentos do homem serão representados por:

u ou U - para cima;  
d ou D - para baixo;  
l ou L - para esquerda;  
r ou R - para direita;

O homem e o cachorro deverão ser representados por ponteiros e os movimentos de ambos deverão ser feito por meio de operação de ponteiros e não utilizando os índices diretamente na matriz. **(se for feita de outra maneira, a questão será anulada)**. Além disso, a checagem de limites da matriz também deve ser feita **através de ponteiros**, sendo vedada a utilização dos índices da matriz nesta checagem.

OBS: Se esta parte: "de operação de ponteiros e não atribuição diretamente ao terreno" não está clara, o que se quer dizer é quer você criará um ponteiro representando o homem e outro o cachorro e então fazer os movimentos através dos mesmos, não sendo podendo utilizar os índices do terreno para movimento. Ou seja, o movimento terá que ser feito usando aritmética de ponteiros.

Use a função srand() e a função rand() para gerar números aleatórios , para os movimentos do cachorro.

3) Marina Maria é um garota muito interessada em matemática, principalmente em relação ao assunto de conjuntos, mas ela recebeu uma tarefa de casa muito complicada e sabendo de seus conhecimentos em programação, pediu sua ajuda para resolver os problemas com um programa em C.

Mas Marina tem um problema: o seu computador tem um problema de memória, então ela pede que você use somente o mínimo de espaço possível, utilizando ponteiros e alocação dinâmica.

Você, como um bom amigo de Marina, deverá fazer um programa que receba:

- Um inteiro N e N inteiros, representando o conjunto 1
- Um inteiro M e M inteiros, representando o conjunto 2

Seu programa deverá criar e preencher mais três vetores:

- Um vetor que represente a diferença entre o conjunto 1 e o conjunto 2
- Um vetor que represente o conjunto complementar do conjunto 2 em relação ao conjunto 1
- Um vetor intersecção que contenha os elementos que estão presentes nos dois conjuntos

Após o cálculo, o seu programa deverá exibir os três vetores

Exemplo:

Entrada :

4

1 2 3 4

3

2 5 7

Saída:

1 3 4

5 7

2

4) Uma função muito útil em editores de texto é o “substituir”, que, recebendo dois textos, substitui o original por um novo em um documento. Você faz parte de uma equipe de programadores que estão desenvolvendo um editor de texto, e foi designado implementar esse recurso.

O seu objetivo é criar uma função que receba como parâmetro três elementos: Um documento de texto, e dois trechos (Original e o novo, respectivamente). A função deve retornar um novo texto, com as devidas modificações e sem desperdício de memória(Ou seja, com um array de tamanho exato ao da string).

Entrada:

O rato roeu a roupa do rei de roma.

//Texto a ser modificado

O rato

//Original

A ratazana

//Modificado

Saída:

A ratazana roeu a roupa do rei de roma.

Nessa questão. É proibido o uso da biblioteca string.h

5) Uma imagem digital é a representação de uma imagem bidimensional usando números binários codificados. Uma das formas de codificar uma imagem colorida é usando o sistema RGB, onde cada pixel é representado por três valores, definindo assim a intensidade das camadas de cores vermelha (Red), verde (Green) e azul (Blue).

Um determinado banco pretende digitalizar todo o seu arquivo de documentos, e chamou você para executar essa tarefa. A tarefa era simples, até você ser avisado que, por questão de gastos para armazenar as versões digitais dos documentos, esses terão que ser convertidos para imagens em preto e branco, para assim, economizar espaço.

O seu trabalho é fazer um programa que receba uma imagem colorida de 24 bits e converta a mesma para uma versão em preto e branco.

Essa transição é feita em dois passos:

1. Converter a imagem em uma equivalente em tons de cinza (Grayscale).
2. Com a versão em grayscale, executar a conversão para preto e branco (B&W)

A conversão de RGB para Grayscale é facilmente feita através da seguinte fórmula:

$$GS = 0.299 * R + 0.587 * G + 0.114 * B$$

Uma das formas de converter uma imagem em grayscale para B&W é através da média da imagem. A partir do valor obtido, os pixels cujo tom de cinza é igual ou menor que a média se torna preto (0) e qualquer pixel com o valor acima se torna branco (1).

A média de uma imagem é dada pela soma dos valores em grayscale de todos os pixels, dividida pelo número de pixels.

A entrada do programa será dada da seguinte forma:

1. Dois inteiros (L e C), representando o número de linhas e colunas de pixels da imagem.
  2. L linhas de 24\*C bits (0 e 1), equivalentes a cada linha da imagem, onde a cada 24 bits se tem um pixel da imagem (8bits para cada uma das camadas de cores RGB, em sequência).
- A saída do programa é uma matriz de L linhas e C colunas, com os valores binários da imagem em preto e branco.

De posse dessas informações sabemos que o programa executa, em sequência, as seguintes ações:

1. Lê toda a imagem, armazenando as strings de bits na memória.
2. Faz a conversão de strings para inteiros, e em sequência, a mudança de base.
3. Utilizando os valores em RGB, converte a imagem para tons de cinza.
4. Usando os valores obtidos na etapa anterior, obtém a versão B&W da imagem e a imprime na tela.

Observações:

1. Na leitura, os bits serão armazenados numa matriz de pixels, onde cada pixel é um vetor de 3 strings(RGB, respectivamente).
2. Depois da conversão das strings para números, a imagem colorida ficará armazenada numa matriz de pixels, onde cada pixel é um vetor de 3 inteiros(RGB).

3. A imagem em tons de cinza ou preto e branco serão armazenadas numa matriz de pixel, onde cada pixel é um unsigned char.

Entrada:

65

[illegible]

Saída:

Media: 127

10101

01010

10101

01010

10101

01010

Entrada:

14 5

[illegible]

Saída:

Media: 118

10000

01000

00111

10011

00011

01111

01111

10011

01011

01110

11110

10110

10100

01100