

INF1005 - PROGRAMAÇÃO 1

LISTA DE EXERCÍCIOS 14

1. Faça um programa para ler uma cadeia de caracteres (máximo 40 caracteres) e outros dois caracteres. Em seguida, o programa deve exibir a cadeia de caracteres substituindo todas as ocorrências do primeiro caractere pelo segundo caractere. Por exemplo, se o usuário digitar "Programacao", e os caracteres 'a' e 'e', o programa deve mostrar "Progremeceo".

O seu programa deve implementar e usar uma função que recebe como parâmetro uma string e dois caracteres (original e novo), e substitui todas as ocorrências do caractere original na string pelo caractere novo. A função deve obedecer ao protótipo:

```
void troca_letra(char str[], char original, char novo);
```

2. Faça um programa para ler uma cadeia de caracteres (máximo 40 caracteres) e em seguida exibir a string invertida. O seu programa deve implementar e usar uma função que receba como parâmetro uma string e a inverta de trás para frente. Por exemplo, ao receber a string "puc-rio", a função deve transformar a string em "oir-cup". A função deve obedecer ao protótipo:

```
void inverte_string(char str[]);
```

3. Faça um programa para ler uma cadeia de caracteres (máximo 40 caracteres) e em seguida exibir somente uma parte dessa string. O seu programa deve implementar e usar uma função chamada `imprime_pedaco` que receba uma string e dois inteiros representando duas posições válidas (inicial e final) da string. A função deve exibir os caracteres da string a partir da posição inicial especificada até a posição final. A função deve obedecer ao protótipo:

```
void imprime_pedaco(char str[], int inicio, int fim);
```

3. Crie um programa que seja capaz de embaralhar uma string S1 com uma string S2 e colocar o resultado em uma string S3. Para embaralhar S1 com S2 é necessário preencher os índices pares de S3 com os elementos de S1 e os ímpares com os elementos de S2 até que os elementos de uma das duas strings termine. Os demais elementos de S3 serão preenchidos com os elementos da string restante.

Exemplo:

```
Digite a string S1: local
Digite a string S2: misterio
Nova string S3: lmoicsatlerio
```

Seu programa deve implementar e usar uma função com o seguinte protótipo:

```
void embaralha_strings(char s1[], char s2[], char s3[]);
```

Seu programa deve pedir ao usuário para digitar as duas strings, chamar a função `embaralha_strings` e exibir as strings obtidas e a string embaralhada.

4. Uma revista de caça-palavras recebeu algumas reclamações dos seus clientes e deseja calcular algumas estatísticas sobre as letras armazenadas na matriz 20x20 das suas atividades de caça-palavras.

Faça um programa que leia de um arquivo texto chamado "letras.txt" as letras do caça-palavras e mostre as seguintes estatísticas:

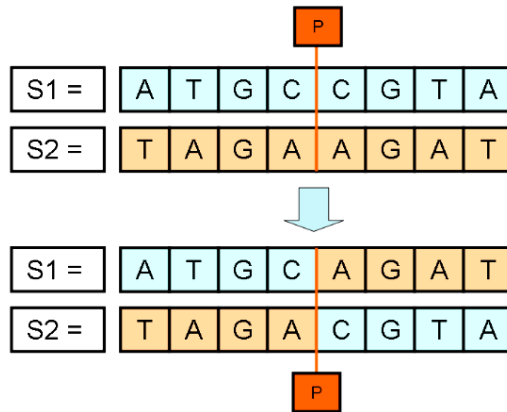
- a) Quantidade de vezes que há letras iguais em posições vizinhas em cada linha;
- b) Quantidade de vezes que a letra 'Z' aparece na matriz;
- c) Média de vogais por coluna;
- d) Quantidade de letras 'Q' na diagonal principal;
- e) Quantidade de letras 'U' acima da diagonal principal;

O seu programa deve implementar e utilizar uma função para cada uma das estatísticas a ser calculada.

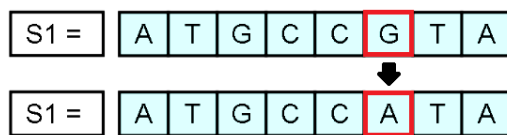
5. Um algoritmo genético é técnica de inteligência artificial que se inspira na forma como a natureza funciona e na teoria da evolução para resolver problemas complexos. Os algoritmos genéticos funcionando mantendo uma população de estruturas que evoluem de forma semelhante à evolução das espécies.

A recombinação e a mutação são os operadores básicos de um algoritmo genético.

Considerando que os indivíduos da população são codificados através de cadeias de caracteres, o operador de recombinação recebe duas strings S1 e S2, sorteia aleatoriamente um ponto de corte em S1 e S2. Em seguida, realizada a troca de informações genéticas de S1 e S2, de forma a gerar dois novos indivíduos filhos. A figura abaixo ilustra o processo de recombinação.



O operador de mutação consiste em sortear aleatoriamente uma posição em um dos indivíduos filhos, e então modificar aleatoriamente o conteúdo daquela posição. A figura abaixo ilustra o processo de mutação.



Construa um programa que implemente e utilize as funções `recombinacao` e `mutacao`:

- A função `recombinacao` recebe como parâmetro 4 strings (S1, S2, S3 e S4), onde S1 e S2 são as strings representado os indivíduos pais e S3 e S4 devem retornar os dois indivíduos filhos gerados com a recombinação de S1 e S2. A função também recebe como parâmetro o tamanho das strings (ambas possuem o mesmo tamanho).
- A função `mutacao` recebe como parâmetro uma string S1 e modifica o conteúdo de S1 realizando o processo de mutação. A função também recebe como parâmetro o tamanho da string S1.

O seu programa deve ler duas strings S1 e S2 (tamanho máximo 20), fazer a recombinação dos dois indivíduos e aplicar uma mutação em um dos indivíduos resultantes. Por ultimo, o programa deve exibir os 4 indivíduos da população (S1, S2, S3 e S4).