

Algoritmos e Programação de Computadores

Lista de Exercícios 4

1. Faça um programa que leia do teclado uma string (possivelmente com espaços) de até 80 caracteres e que então salve a inversa desta string em uma nova string. Refaça o programa de tal forma que não seja utilizado nenhum vetor adicional! Ou seja, devemos computar a inversa no próprio vetor original da string lida.
2. Faça um programa que leia do teclado uma string (possivelmente com espaços) de até 80 caracteres e que então salve a string lida em uma nova removendo-se os espaços.

Exemplo de entrada:

```
Out    of the night that    covers me
```

e vetor resultante:

```
Outofthenightthatcoversme
```

Refaça o programa de tal forma que não seja utilizado nenhum vetor adicional! Ou seja devemos deixar a string sem espaços no próprio vetor original da string lida sem o uso de nenhum outro vetor auxiliar.

3. Faça um programa que leia do teclado uma string (possivelmente com espaços) de até 80 caracteres e que então salve a string lida em uma nova removendo-se os espaços **extras** entre as palavras.

Exemplo de entrada:

```
Out    of the night that    covers me
```

e vetor resultante:

```
Out of the night that covers me
```

Refaça o programa de tal forma que não seja utilizado nenhum vetor adicional! Ou seja devemos deixar a string sem espaços extras no próprio vetor original da string lida sem o uso de nenhum outro vetor auxiliar.

4. Escreva um programa que leia duas palavras do teclado e determina se a segunda é um *anagrama* da primeira. Uma palavra é um anagrama de outra se todas as letras de uma ocorrem na outra, *em mesmo número, independente da posição*. Exemplos: ROMA, MORA, ORAM, AMOR, RAMO são anagramas entre si.
5. Faça um programa que leia duas strings e elimine, da segunda string, todas as ocorrências dos caracteres da primeira string.

6. Faça um programa que leia um texto T (com espaços) e uma palavra p do teclado. Em seguida o programa deverá imprimir todas as posições onde ocorrem a palavra p em T .
Se por exemplo $T = \text{"duas bananas e 4 abacates. Nao haverá mais bananas."}$, e $p = \text{"bananas"}$, então o programa deveria imprimir 5 e 43.

7. Escreva um programa que lê uma string de até 50 caracteres, e imprime "Palindromo" caso a string seja um palindromo e "Nao Palindromo" caso contrário. OBS: Um palindromo é uma palavra ou frase, que é igual quando lida da esquerda para a direita ou da direita para a esquerda (assuma que só são usados caracteres minúsculos e sem acentos. Espaços em brancos devem ser descartados). Exemplo de palindromo: saudavel leva duas.

8. Faça um programa que leia duas palavras e verifique se uma delas pode ser obtida por meio da remoção de letras da outra. A ordem das letras não pode ser alterada.

Por exemplo:

moda é uma subsequência em *moradia*

cereja é uma subsequência em *cerveja*

9. Historicamente César foi o primeiro a codificar mensagens. Ele reorganizava o texto de suas mensagens de maneira que o texto parecia não ter sentido. Cada mensagem sempre possuía uma contagem de letras cujo total equivalia a um quadrado perfeito, dependendo de quanto César tivesse que escrever. Assim, uma mensagem com 16 caracteres usava um quadrado de quatro por quatro; se fossem 25 caracteres, seria cinco por cinco; 100 caracteres requeriam um quadrado de dez por dez, etc. Seus oficiais sabiam que deviam transcrever o texto preenchendo as casas do quadrado sempre que uma mensagem aleatória chegasse. Ao fazerem isso, podiam ler a mensagem na vertical e seu sentido se tornaria claro.

Escreva um programa que lê o tamanho de uma string e em seguida uma string. Depois o programa escreve a mensagem decifrada.

Exemplo:

36

MEEUMOCSHMSC1T*AGUOA***L2****T*****A

Esta mensagem pode ser transcrita em um quadrado perfeito 6x6.

M	E	E	U	M	O
C	S	H	M	S	C
1	T	*	A	G	U
0	A	*	*	*	L
2	*	*	*	*	T
*	*	*	*	*	A

Lendo cada coluna da matriz (desconsiderando o caractere '*'), a saída deve ser:

MC102 ESTA EH UMA MSG OCULTA.