

## *Disciplina: Algoritmos II*

### *Professor: Adilso Nunes de Souza*

#### **Orientações:**

**-Realizar os exercícios propostos abaixo, para entregar compacte todos os arquivos .cpp em um único diretório com o nome do aluno e o número da atividade (nome\_do\_aluno\_Atividade\_10) e realize a entrega na atividade no classroom.**

#### **Lista de exercícios 10**

**12** – Crie um programa que receba um texto qualquer do teclado e implemente uma função que utilizando aritmética de ponteiro realize as seguintes ações:

- Transformar todos os caracteres do texto para maiúsculo
- Remover os caracteres especiais existentes no texto
- Remover os espaços em branco.
- Mostrar o texto resultante.

**13** - Escreva uma função que receba como parâmetro um array de inteiros com N valores, e determina o maior elemento do array e o número de vezes que este elemento ocorreu no array. Por exemplo, para um array com os seguintes elementos: 5, 2, 15, 3, 7, 15, 8, 6, 15, a função deve fornecer para o programa que a chamou o valor 15 e o número 3 (indicando que o número 15 ocorreu 3 vezes). A função deve ser do tipo void, utilize ponteiros para fornecer as informações solicitadas

**14** - Escreva um programa que utilizando ponteiro e alocação dinâmica de memória, manipule um vetor de inteiros de 10 posições, o vetor deverá ser preenchido de forma pseudo-aleatória com valores entre 30 e 55, mostrar o vetor gerado na tela, realizar a troca dos valores no vetor de tal forma que o primeiro passa a ser o último, o segundo o penúltimo e assim sucessivamente. Mostre o vetor na tela após as trocas, utilize aritmética de ponteiro para percorrer o vetor.

**15** – Escreva um programa que leia um valor inteiro N, sendo  $3 \leq N \leq 9$ . Em seguida, imprima um padrão triangular de números, conforme os exemplos abaixo.

										N = 9									
										N = 5									
										N = 4									
										1									
										1 2									
										1 2 3									
										1 2 3 4									
										1 2 3 4 5									
										1 2 3 4 5 6									
										1 2 3 4 5 6 7									
										1 2 3 4 5 6 7 8									
										1 2 3 4 5 6 7 8 9									
1																			
1 2																			
1 2 3																			
1 2 3 4																			
1 2 3 4 5																			
1 2 3 4 5 6																			
1 2 3 4 5 6 7																			
1 2 3 4 5 6 7 8																			
1 2 3 4 5 6 7 8 9																			