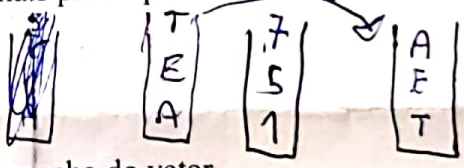
Curso de Ciência da Computação
Estrutura de Dados I – 2023.2
Prof. Anselmo Cardoso de Paiva

1. Faça um algoritmo que recebe duas matrizes nxn armazenadas nos vetores v1 e v2 e, calcula uma nova matriz a ser armazenada no vetor v3 que corresponde a multiplicação da matriz armazenada no vetor v1 pela matriz armazenada no vetor v2. Considere que o vetor v3 já está devidamente alocado.
   int MultiplicaMatrizes (int *v1, int *v2, int *v3, int n)

2. Faça um algoritmo que recebe uma fila implementada como um vetor circular e promove o n-esimo elemento da fila (caso ele exista) colocando-o na primeira posição. Caso a fila tenha menos que n elementos coloca o último elemento na primeira posição da fila.
   int PromoveElementoFila (Queue *q, int n)

3. Faça um algoritmo que recebe um vetor de caracteres (string) formado por uma sequência alternada de letras e dígitos (primeiro caractere é uma letra), faça um algoritmo usando o TAD PILHA que retorne um vetor de caracteres (string) no qual as letras são mantidas na sequência original, seguidas dos números colocados na ordem inversa. O Algoritmo deve usar uma ou mais pilhas pra resolver o problema, usando as funções do TAD Pilha.
   Exemplos:
   A 1 E 5 T 7 W 8 G → A E T W G 8 7 5 1
   G 3 C 9 H 4 Q 6 → G C H Q 6 4 9 3
   char *InverteNumeros (char *s, int n) // n é o tamanho do vetor



4. Faça um algoritmo que recebe uma pilha armazenada em um vetor, um valor (chave), e uma função de comparação, e remove respeitando a disciplina de acesso da pilha todos os elementos até encontrar um com chave menor que o valor da chave recebida. Não pode usar pops e push, e deve obedecer a disciplina de acesso da pilha. É uma função interna do TAD Pilha.
   int RemoveMaioresQueKey (Stack *s, void *key, int (*cmp) (void *, void *))
   OBS: cmp (a,b) retorna TRUE se a <b e False caso contrário