

A universidade está comprometida em preservar a reputação do seu diploma. Para garantir que cada diploma outorgado possua o significado que ele realmente representa, nesta disciplina exige-se que o acadêmico:

- Entregue um trabalho original, isto é, desenvolvido de forma integral e individual pelo próprio acadêmico.
- Não utilize uma solução parcial ou completa obtida de qualquer outra fonte (outra pessoa, Internet, etc).
- Não discuta seu trabalho com outros colegas.
- Proteja seu trabalho e evite que ele seja roubado. Não compartilhe seu trabalho com ninguém.
- Não entrega um trabalho que não é seu.

O não cumprimento destas regras caracteriza desonestade acadêmica e a sua nota pode ser zerada. Além disso, penalidades como suspensão ou expulsão podem ser impostas pelo departamento de Engenharia de Computação.

Nesta atividade você demonstrará domínio básico sobre algoritmos: variáveis, tipos de dados, expressões, estruturas condicionais, de repetição, arrays e funções. O algoritmo a ser desenvolvido realizará diversos tipos de manipulações em arrays de caracteres e números inteiros.

Para realizar esta tarefa você deve criar um *array* para armazenar 13 números inteiros e outro para armazenar 13 caracteres. O *array* de inteiros deve ser inicializado com o valor 0 (zero), enquanto o *array* de caracteres deve ser inicializado com 'A' (letra A).

Todas as operações que envolvem trabalhar com o *array* inteiro devem ser feitas em cima do mesmo *array* de inteiro. Da mesma forma, todas as operações que envolvem trabalhar com o *array* de caracteres devem ser feitas em cima do mesmo *array* de caracteres.

Ao ser executado o algoritmo imprime um menu com as opções listadas abaixo na tela. As operações que são marcadas com (i) devem utilizar o *array* inteiro, e as operações marcadas com (c) devem ser feitas com o *array* de caracteres.

Assim que o usuário decidir a operação que gostaria de executar, ele digita o número da opção desejada e o algoritmo procede para executar aquela tarefa específica. Quando ela terminar, o algoritmo volta a imprimir o menu na tela e permitir que o usuário selecione uma nova opção. Desta forma, o algoritmo só termina quando o usuário escolher a opção 0, encerrando a execução do algoritmo.

É importante observar que se o usuário digitar uma opção inexistente, o algoritmo deve imprimir na tela a mensagem "Opção inválida!" e apresentar o menu novamente para que o usuário selecione uma outra opção.

Por favor selecione uma das opções abaixo:

- 0 - Sair
- 1 - (i) Imprimir o array
- 2 - (i) Inicializar o array a partir de valores do usuário
- 3 - (i) Imprimir um elemento a partir de seu índice
- 4 - (i) Imprimir os números pares contidos no array
- 5 - (i) Imprimir a média aritmética dos valores do array
- 6 - (i) Substituir todas as ocorrências de um valor por outro do usuário
- 7 - (i) Elevar cada elemento do array ao cubo
- 8 - (i) Imprimir os elementos que são resultados de uma potência de 3
- 9 - (i) Deslocar os elementos duas casas para a direita
- 10 - (i) Imprimir a quantidade de dígitos de cada elemento do array
- 11 - (c) Imprimir o array
- 12 - (c) Inicializar o array a partir de valores do usuário
- 13 - (c) Imprimir o número de ocorrências de um determinado caractere
- 14 - (c) Imprimir as vogais presentes no array
- 15 - (c) Imprimir as consoantes presentes no array
- 16 - (c) Imprimir os caracteres numéricos (0-9) presentes no array
- 17 - (c) Inverter o tamanho da caixa das letras (a-z, A-Z) do array
- 18 - (c) Informar se o array está balanceado
- 19 - (c) Informar se o array possui os mesmos valores de trás pra frente
- 20 - (c) Informar o caractere que possui o maior número de ocorrências

Sua implementação deve demonstrar no mínimo:

- Uma estrutura condicional simples;
- Uma estrutura condicional composta;
- Uma estrutura condicional de seleção múltipla (escolha-caso);
- Uma estrutura condicional aninhada;
- Uma estrutura de repetição para-até-faça;
- Uma estrutura de repetição enquanto-faça;
- Uma estrutura de repetição repita-até;
- 20 implementações de funções.

A lista abaixo apresenta os detalhes de funcionamento de cada operação do menu:

0 – Sair

Esta opção deve terminar a execução do algoritmo.

1 – (i) Imprimir o array

Conteúdo do array: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233]

Saída:

1 1 2 3 5 8 13 21 34 55 89 144 233

2 – (i) Inicializar o array a partir de valores do usuário

Esta operação muda o conteúdo do array permanentemente.

Conteúdo do array: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Entrada:

1 1 2 3 5 8 13 21 34 55 89 144 233

Conteúdo do array após a operação:

1 1 2 3 5 8 13 21 34 55 89 144 233

3 – (i) Imprimir um elemento a partir de seu índice

Conteúdo do array: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233]

Entrada:

6

Saída:

13

4 – (i) Imprimir os números pares contidos no array

Conteúdo do array: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233]

Saída:

2 8 34 144

5 – (i) Imprimir a média aritmética dos valores do array

Conteúdo do array: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233]

Saída:

46.8

6 – (i) Substituir todas as ocorrências de um valor por outro do usuário

Esta operação muda o conteúdo do array permanentemente.

Conteúdo do array: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233]

Entrada:1
999

Conteúdo do array após a operação:

999 999 2 3 5 8 13 21 34 55 89 144 233

7 – (i) Elevar cada elemento do array ao cubo

Esta operação muda o conteúdo do array permanentemente.

Conteúdo do array: [1, 3, 5, 7, 9, 11, -99, 12, 10, 8, 6, 4, 2]

Conteúdo do array após a operação:

1 27 125 343 729 1331 -970229 1728 1000 512 216 64 8

8 – (i) Imprimir os elementos que são resultados de uma potência de 3

Conteúdo do array: [1, 2, 27, 4, 81, 11, -99, 243, 10, 3, 9, 6561, 64]

Saída:

1 27 81 243 3 9 6561

9 – (i) Deslocar os elementos duas casas para a direita

Esta operação muda o conteúdo do array permanentemente.

Conteúdo do array: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233]

Conteúdo do array após a operação:

144 233 1 1 2 3 5 8 13 21 34 55 89

10 – (i) Imprimir a quantidade de dígitos de cada elemento do array

Conteúdo do array: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233]

Saída:

1 1 1 1 1 1 2 2 2 2 2 3 3

11 – (c) Imprimir o array

Conteúdo do array: ['a', 's', 'D', 'F', '9', '5', '1', '0', '?', '\$', 'X', 'y', 'Z']

Saída:

a s D F 9 5 1 0 ? \$ X y Z

12 – (c) Inicializar o array a partir de valores do usuário

Esta operação muda o conteúdo do array permanentemente.

Conteúdo do array: ['A', 'A']

Entrada:

a s D F 9 5 1 0 ? \$ X y Z

Conteúdo do array após a operação:

a s D F 9 5 1 0 ? \$ X y Z

13 – (c) Imprimir o número de ocorrências de um determinado caractere

Conteúdo do array: [‘a’, ‘Z’, ‘D’, ‘F’, ‘9’, ‘5’, ‘1’, ‘0’, ‘?’, ‘\$’, ‘X’, ‘y’, ‘Z’]

Entrada:

z

Saída:

2

14 – (c) Imprimir as vogais presentes no array

Conteúdo do array: [‘a’, ‘s’, ‘D’, ‘F’, ‘9’, ‘5’, ‘1’, ‘0’, ‘?’, ‘\$’, ‘X’, ‘y’, ‘Z’]

Saída:

a

15 – (c) Imprimir as consoantes presentes no array

Conteúdo do array: [‘a’, ‘s’, ‘D’, ‘F’, ‘9’, ‘5’, ‘1’, ‘0’, ‘?’, ‘\$’, ‘X’, ‘y’, ‘Z’]

Saída:

D F X y Z

16 – (c) Imprimir os caracteres numéricos (0-9) presentes no array

Conteúdo do array: [‘a’, ‘s’, ‘D’, ‘F’, ‘9’, ‘5’, ‘1’, ‘0’, ‘?’, ‘\$’, ‘X’, ‘y’, ‘Z’]

Saída:

9 5 1 0

17 – (c) Inverter o tamanho da caixa das letras (a-z, A-Z) do array

Esta operação muda o conteúdo do array permanentemente.

Conteúdo do array: [‘a’, ‘s’, ‘D’, ‘F’, ‘9’, ‘5’, ‘1’, ‘0’, ‘?’, ‘\$’, ‘X’, ‘y’, ‘Z’]

Conteúdo do array após a operação:

A S d f 9 5 1 0 ? \$ x Y z

18 – (c) Informar se o array está balanceado

Conteúdo do array: [‘a’, ‘s’, ‘D’, ‘F’, ‘9’, ‘5’, ‘1’, ‘0’, ‘?’, ‘\$’, ‘X’, ‘y’, ‘Z’]

Esta operação divide o array exatamente no meio. Ela conta quantas letras (a-z A-Z) existem nos 6 primeiros elementos e depois conta quantas existem nos últimos 6 elementos. O array está balanceado quando estes números forem iguais.

Saída:

O array não está balanceado: a primeira metade tem 4 letras e a segunda tem 3 letras.

19 – (c) Informar se o array possui os mesmos valores de trás pra frente

Conteúdo do array: [‘a’, ‘s’, ‘D’, ‘F’, ‘9’, ‘5’, ‘1’, ‘5’, ‘9’, ‘F’, ‘D’, ‘s’, ‘a’]

Saída:

O array possui os mesmos valores de trás pra frente.

20 – (c) Informar o caractere que possui o maior número de ocorrências

Conteúdo do array: [‘a’, ‘z’, ‘Z’, ‘F’, ‘9’, ‘5’, ‘1’, ‘3’, ‘8’, ‘g’, ‘p’, ‘Z’, ‘b’]

Saída:

O caractere com mais ocorrências é o Z, possuindo 2.