

1. Sendo o vetor V igual a:

2	6	8	3	10	9	1	21	33	14
0	1	2	3	4	5	6	7	8	9

Considere as variáveis $x=2$ e $y=4$. Escreva o elemento obtido ou "ERRO" para acessar uma posição:

- | | | | |
|-------------------|------------------|--------------|-----------------|
| a) $v[x+1]$ | b) $v[x+2]$ | c) $v[x+3]$ | d) $v[y*1]$ |
| e) $v[x*3]$ | f) $v[x*2]$ | g) $v[x*5]$ | h) $v[v[x+y]]$ |
| i) $v[x+y]$ | j) $v[8 - V[2]]$ | k) $v[v[4]]$ | l) $v[v[v[7]]]$ |
| m) $v[v[1]*v[4]]$ | n) $v[x+4]$ | | |

2. Escreva um programa em linguagem C para gerar, aleatoriamente, um vetor com 1000 números inteiros. Calcular e apresentar as informações:
- A média dos números;
 - A média dos números pares;
 - A média dos números ímpares;
 - A quantidade de números pares;
 - A quantidade de números ímpares.
3. Escreva um programa em linguagem C para que usuário forneça 10 valores inteiros. O programa deve permitir entradas de valores maiores que 0. Os primeiros cinco valores devem ser armazenados em um vetor A e os demais em um vetor B. Em seguida, o programa deve efetuar a troca dos valores existentes em A e B, considerando a regra: o valor da última posição do vetor A é trocado com o valor da primeira posição do vetor B, o valor da penúltima posição do vetor A é trocado com o valor da segunda posição do vetor B e assim sucessivamente. Esse processo deve ser realizado por meio de estruturas de repetições.
4. Escreva um programa em linguagem C para que usuário forneça cinco valores inteiros. O programa deve permitir entradas de valores maiores que 0. Os primeiros cinco valores devem ser armazenados em um vetor A e os demais em um vetor B. Em seguida, o programa deve efetuar a multiplicação entre os valores disponíveis em A e B, armazenando o resultado em um terceiro vetor C. As posições devem ser multiplicadas de maneira invertida, ou seja, a primeira posição do vetor A deve ser multiplicada pela última posição do vetor B. O resultado deve ser armazenado na última posição do vetor C. Esse processo deve ser realizado por meio de estruturas de repetições.
5. Escreva um programa para simular as operações de uma pilha (First in last out - FILO), com 10 posições. O primeiro elemento a ser empilhado é o último a ser retirado da pilha. Este tipo de estrutura é comumente utilizado para gerenciar chamadas de funções, por exemplo. O programa deve ter as operações de inserção e remoção. O processo de remoção não é físico. Uma remoção deve ser controlada por uma indicação lógica.
6. Escreva um programa para simular as operações de uma fila (FIFO), ou seja, o primeiro elemento a entrar na fila é o primeiro a ser retirado da fila. O programa deve permitir uma fila com 10 posições. Uma lista FIFO é comumente utilizada para gerenciar processos em que a ordem de chegada é a que deve ser utilizada para o atendimento, por exemplo, como ocorre em uma fila de impressão. O programa deve ter as operações de inserção e remoção. O processo de remoção não é físico. Uma remoção deve ser controlada por uma indicação lógica.

8. Escreva um programa para controlar o estoque de uma loja que trabalha com cinco tipos de produtos. O algoritmo deverá permitir que o usuário escolha uma das seguintes opções:
Aumentar estoque
Baixar estoque
Visualizar estoque de um produto
Visualizar estoque total
Aumentar preço
Diminuir preço
Trocar produto
Sair
9. Escreva um programa para calcular a conta dos clientes de uma lanchonete que oferece três tipos de lanches, três tipos de pizzas e três tipos de bebidas. A capacidade da lanchonete é de 10 clientes. O algoritmo deve permitir a escolha no menu das seguintes opções:
Menu Principal: [1] – Lanches [2] – Pizzas [3] – Bebidas [4] – Fechar Conta [5] – Fechar Caixa [6] [7] – Sair
A opção 6 deve apresentar uma lista dos totais de produtos que foram consumidos, bem como o total em R\$. Os preços (R\$) são: lanche (15,00); Pizzas (87,00); Bebidas (8,00).
10. Escreva um programa para controlar a conta de clientes de um banco. Para fins ilustrativos, o programa deve trabalhar com cinco clientes. A conta de cada cliente deve ter os seguintes dados: número, CPF, tipo (p-poupança ou c-corrente) e saldo. O algoritmo deve permitir as seguintes operações: cadastrar conta, buscar conta pelo cpf, buscar todas as contas de um cliente (pelo cpf), efetuar depósito, efetuar retirada, mostrar saldos de todas as contas. Se possível, o sistema deve ter a opção extrato.
11. Escreva um programa para ler um vetor com 500 posições e, em seguida, permitir que o usuário escolha algumas operações:
Ordenar elementos em ordem crescente
Ordenar elementos em ordem decrescente
Imprimir todo o vetor
Imprimir as posições de um intervalo especificado pelo usuário
Digite <s> ou <S> para encerrar
12. Dado um vetor VIN de 10 elementos inteiros, criar um vetor VAI de 30 elementos, sendo que VAI[0], VAI[1] e VAI[2] recebe o valor de VIN[0] e assim por diante. Escrever um programa que leia VIN e imprima VAI.
13. Dado um vetor A de tamanho 5, capaz de armazenar números inteiros, escreva um programa para obter a maior diferença entre dois elementos consecutivos desse vetor. Imprimir a maior diferença e os índices dos respectivos elementos.
14. Faça um programa que leia um vetor X capaz de armazenar cinco números inteiros. Em seguida, o programa deve dividir todos os elementos contidos em X pelo maior valor do vetor. Mostre o vetor antes e após os cálculos.

15. Elaborar um programa para calcular o reajuste de salário de uma empresa com 10 funcionários. O usuário deve responder se deseja continuar ou não a execução do algoritmo. Considere que o funcionário deverá receber um reajuste de 15%, caso seu salário seja menor que R\$ 500,00. Se o salário for maior ou igual a R\$ 500,00, mas menor ou igual a R\$ 1000,00, o reajuste será de 10%. Se salário for maior que R\$ 1000,00, o reajuste deverá ser de 5%. Apresentar uma lista no seguinte formato:

Lista de Salários dos Funcionários da Empresa AB			
Código	Nome	Salário Base	Salário Reajustado
		Total:	

16. Apresentar os resultados de uma tabuada de um número qualquer (digitado pelo usuário). A tabuada deve ser escrita no seguinte formato: multiplicando x multiplicador = resultado. (Ex. 2 x 2 = 4). Todos os valores devem ser calculados e apresentados ao final, seguindo o formato indicado.
17. Faça um programa que leia um vetor A capaz de armazenar 10 valores. Em seguida, um vetor B deve receber os valores A, excluindo-se números nulos e negativos. Apresente os vetores.
18. Escreva um programa em linguagem C para gerar, aleatoriamente, 10 números inteiros. Os valores devem ser armazenados um vetor com 11 posições. Em seguida, o programa deve solicitar duas entradas, um elemento e um índice. O índice deve ser um valor entre 0 e 9. O elemento informado deve ser inserido na posição especificada, deslocando todos os elementos seguintes. Imprimir o resultado.
19. Escreva um programa que leia dois vetores A e B com 10 posições. Em seguida, o programa deve realizar a multiplicação entre os elementos (com o mesmo índice) contidos nos vetores. O resultado deve ser armazenado em um vetor C. Mostre o vetor resultante.
20. Seja A e B dois vetores contendo 10 elementos inteiros. Fazer um programa para:
- ler A e B.
 - Calcular a soma dos elementos de A.
 - Calcular a soma dos elementos de B.
 - Obter o vetor C, que é a soma dos vetores A e B.
 - Obter o vetor D, subtraindo B de A.
21. Um hotel cobra R\$ 300,00 por diária e mais uma taxa adicional de serviços. Se a diária for menor que 15, a taxa é de R\$20,00. Se o número de diárias for igual a 15, a taxa é de R\$14,00. Se o número for maior que 15, a taxa é de R\$ 12,00. Considere que há 200 hóspedes e que para cada um existe um registro (quadro abaixo). Faça um algoritmo para armazenar os dados e ao final apresentar: dados pessoais e o total a pagar de cada hóspede; o total ganho pelo hotel e total de diárias.

Lista de Hóspedes						
Código	Nome	Endereço	Fone	E-mail	Número de diárias	Conta (Total)

Total:		
---------------	--	--

22. Escreva um programa em linguagem C para que usuário digite um nome. A entrada deve ser armazenada em uma variável *nome*, declarada com 40 caracteres. Em seguida, o programa deve percorrer a string *nome* e apresentar cada letra em uma linha. O programa deve percorrer a string somente até a última letra do nome. Ao final, o programa deverá apresentar também o total de algarismos existentes.
23. Escreva um programa em linguagem C para que usuário digite um nome e um sobrenome, sendo esses armazenados em variáveis *nome* e *sobrenome*, respectivamente. As strings têm tamanhos máximos de 40 caracteres. Em seguida, o programa deve percorrer as strings, concatenar as entradas e armazenar o resultado em uma string nomeada *resultado*, declarada com tamanho 80. Ao final, o programa deverá apresentar o resultado da concatenação com o total de algarismos existentes.
24. Escreva um programa para realizar apresentar a tabuada de um número informado pelo usuário, seguindo formato: multiplicando x multiplicador = resultado. Os resultados desse processo devem ser armazenados em um vetor *res*. Apresentar *res*.
25. Elaborar um programa para armazenar a série de *Fibonacci*, em um vetor *fib*, até o décimo quinto termo. A série é formada pela sequência 1, 1, 2, 3, 5, 8, 13, 21, 34, ..., etc. Esta série é caracterizada pela soma de um termo posterior com o seu anterior subsequente. Apresentar o resultado armazenado em *fib*.

Exercícios envolvendo manipulação de string (vetores do tipo char).

26. Escreva um programa em linguagem C para reproduzir tarefas executadas pelas funções: *strcat*, *strcpy* e *strlen*. O usuário deve fornecer duas entradas, armazenadas em *str1* e *str2*. As strings são vetores do tipo char com 40 posições. As entradas de dados devem ser por meio da função *scanf*. O controle do processamento deve ocorrer via análise do '\0'.
27. Escreva um programa que leia uma string e imprima uma tabela com o número de ocorrências de cada caractere na string. O controle do processamento da string deve ocorrer via análise do '\0'.
28. Escreva um programa que indique se uma string é ou não um palíndromo. O controle do processamento deve ocorrer via análise do '\0'.
29. Escreva um programa para ler 10 nomes. Os nomes devem ser armazenados em dois vetores de strings (*str1* e *str2*), cinco nomes em cada vetor. O tamanho máximo de cada string deve ser 60 caracteres. O programa deve mostrar *str1* e *str2*. Em seguida, a string *str3* deve armazenar os nomes contidos em *str1* e *str2*. Apresentar a string *str3*.