

Algoritmos com Vetores – 1.01

1. ☺ Fazer um algoritmo que leia dez números e escreva-os na ordem contrária à ordem de leitura. Exemplo:

| | | | | | | | | | | | | | | | | | | | | | |
|-----|---|----|---|---|----|---|----|----|---|----|----------|----|---|----|----|---|----|---|---|----|---|
| lê: | 7 | 40 | 3 | 9 | 21 | 0 | 63 | 31 | 7 | 22 | escreve: | 22 | 7 | 31 | 63 | 0 | 21 | 9 | 3 | 40 | 7 |
|-----|---|----|---|---|----|---|----|----|---|----|----------|----|---|----|----|---|----|---|---|----|---|

2. ☺ Fazer um algoritmo que leia dez números inteiros armazenando-os em um vetor e escreva primeiramente todos os números pares lidos e após todos os ímpares. Exemplo:

| | | | | | | | | | | | | | | | | | | | | | |
|-----|---|----|---|---|----|---|----|----|---|----|----------|----|---|----|---|---|---|----|----|----|---|
| lê: | 7 | 40 | 3 | 9 | 21 | 0 | 63 | 31 | 7 | 22 | escreve: | 40 | 0 | 22 | 7 | 3 | 9 | 21 | 63 | 31 | 7 |
|-----|---|----|---|---|----|---|----|----|---|----|----------|----|---|----|---|---|---|----|----|----|---|

3. ☺ Fazer um algoritmo que leia trinta números reais armazenando-os em um vetor e após escreva a posição de cada número menor que zero desse vetor. Exemplo:

| | | | | | | | | | | | | | | | | | |
|----------|---------------------|---|------|-----|------|------|----|------|------|------|-----|---|-----|-----|-----|-----|-----|
| lê: | 5,1 | 0 | -3,6 | 4,1 | -2,5 | -1,3 | -4 | 1,39 | -128 | -6,9 | 8,2 | 9 | 154 | -88 | 6,4 | 7,1 | ... |
| escreve: | 3 5 6 7 9 10 14 ... | | | | | | | | | | | | | | | | |

4. ☺ Fazer um algoritmo que leia vinte números inteiros armazenando-os em um vetor, a seguir, troque o conteúdo da primeira posição do vetor com o conteúdo da décima primeira posição, o conteúdo da segunda posição com o da décima segunda, e assim, sucessivamente até trocar o conteúdo da décima com o da vigésima e escreva o vetor após estas modificações.

5. ☺ Fazer um algoritmo que leia vinte números inteiros armazenando-os em um vetor, a seguir, troque o conteúdo da primeira posição do vetor com o conteúdo da segunda posição, o conteúdo da terceira posição com o da quarta, e assim, sucessivamente até trocar o conteúdo da décima nona com o da vigésima e escreva o vetor após estas modificações.

6. ☺ Fazer um algoritmo que leia cem números reais armazenando-os em um vetor. Após desloque todos os elementos deste vetor uma posição a esquerda, colocando na última posição o elemento que estava na primeira. Repetir isso até que o maior elemento do vetor fique na primeira posição do mesmo. Ex (com seis elementos):

| | | | | | | |
|-----------------------|-----|-----|-----|-----|-----|-----|
| valores lidos: | 6,0 | 2,5 | 1,9 | 8,2 | 2,7 | 5,5 |
| após 1º deslocamento: | 2,5 | 1,9 | 8,2 | 2,7 | 5,5 | 6,0 |
| após 2º deslocamento: | 1,9 | 8,2 | 2,7 | 5,5 | 6,0 | 2,5 |
| após 3º deslocamento: | 8,2 | 2,7 | 5,5 | 6,0 | 2,5 | 1,9 |

7. ☺ Fazer um algoritmo que leia cem números reais e escreva o percentual de números que são maiores que a média dos mesmos.

8. ☺ Fazer um algoritmo que armazene em um vetor os quarenta primeiros números da série de Fibonacci e após escreva os elementos deste vetor.

| | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|----|----|----|----|----|-----|-----|-----|
| posição: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | ... |
| conteúdo: | 0 | 1 | 1 | 2 | 3 | 5 | 8 | 13 | 21 | 34 | 55 | 89 | 144 | 233 | ... |

9. ☺ Fazer um algoritmo que leia quinze número reais, escreva a média destes números e o menor número maior que a média (o que acontece se todos os números forem iguais?). Exemplo:

| | | | | | | | | | | | | | | | |
|----------|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| lê: | 9,5 | 1,2 | 1,3 | 3,0 | 1,5 | 2,0 | 8,5 | 4,5 | 9,0 | 0,4 | 9,3 | 7,5 | 7,2 | 1,9 | 1,5 |
| escreve: | Média=4,55 Menor número maior que a média=7,2 | | | | | | | | | | | | | | |

10. ☺ Fazer um algoritmo que leia uma quantidade indeterminada de números até que sejam informados dez números distintos e escreva esses dez números. Exemplo:

| | | | | | | | | | | | | | | | |
|----------|----|----|---|----|----|----|----|----|----|----|----|----|---|---|----|
| lê: | 27 | 12 | 5 | 15 | 12 | 20 | 5 | 6 | 45 | 23 | 90 | 12 | 6 | 6 | 33 |
| escreve: | 27 | 12 | 5 | 15 | 20 | 6 | 45 | 23 | 90 | 33 | | | | | |

11. ☺ Fazer um algoritmo que coloque em um vetor os quinhentos primeiros números primos e após escreva os elementos deste vetor. Para identificar se um número é primo, divida-o pelos primos que já estão no vetor. Pensar se é necessário dividir por todos os números já gerados.

12. ☹ Fazer um algoritmo que leia cem números inteiros e escreva o número que mais vezes aparece.
13. ☹ Fazer um algoritmo que leia números inteiros, armazenando os números distintos em um vetor V, até que seja informado um número negativo ou sejam armazenados dez números distintos. Após o algoritmo deve fazer o mesmo para armazenar números distintos em um vetor W, e então, ler uma opção e caso essa for:
- 1 - Efetuar a operação de união de conjuntos, armazenando o resultado no vetor R.
 - 2 - Efetuar a operação de intersecção de conjuntos, armazenando o resultado no vetor R.
 - 3 - Efetuar a operação de subtração de conjuntos (V-W), armazenando o resultado no vetor R.
 - 4 - Efetuar a operação de subtração de conjuntos (W-V), armazenando o resultado no vetor R.
- Finalmente, escreva o resultado da operação selecionada acima, ou seja, os números que estão no vetor R.
14. ☹ Em uma disciplina, como resultado final, cada aluno obtém um conceito, que é um número inteiro de zero até quatro. Fazer um algoritmo que leia a quantidade de alunos com conceito zero, com conceito um, com conceito dois, com conceito três e com conceito quatro, armazenando cada uma destas quantidades em um vetor [0..4], e mostre um histograma horizontal conforme exemplo abaixo:

| | | | | | |
|---------------------------------|--|----|---|----|---|
| vetor com as quantidades lidas: | 3 | 12 | 7 | 16 | 5 |
| histograma: | <pre> 0 *** 1 ***** 2 ***** 3 ***** 4 ***** </pre> | | | | |

15. ☹ Fazer um algoritmo que leia notas (inteiras de 0 a 10) de alunos até que seja informada uma nota menor que zero e mostre um histograma vertical correspondente ao número de ocorrências de cada nota. Exemplo:

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----|---|---|----|---|---|---|---|---|----|
| notas: | 3 | 6 | 9 | 0 | 0 | 6 | 7 | 8 | 0 | 9 | 8 | 3 | 6 | 6 | 8 | 10 | 9 | 5 | 10 | 2 | 8 | 5 | 8 | 5 | -1 |
| histograma: | <pre> * * * * * * * * * * * * * * * * * 0 1 2 3 4 5 6 7 8 9 10 </pre> | | | | | | | | | | | | | | | | | | | | | | | | |

16. ☹ Fazer um algoritmo que leia um número inteiro positivo, e escreva o mesmo convertido para binário. Para transformar um número decimal em binário pode-se dividir o número por dois e ir sucessivamente dividindo os quocientes resultantes da divisão anterior por dois até que ele seja igual a zero, e então, escrever os restos obtidos nestas divisões em ordem do último resto obtido para o primeiro. Máximo 32 dígitos binários.
Exemplo: lê: 37 escreve: 1 0 0 1 0 1

17. ☹ Pode-se determinar todos os números primos até um limite superior N pelo crivo de Erastóstenes. Nesse método, inicialmente supõe-se que todos os números entre 2 e N são primos, e a seguir para cada um desses números elimina-se todos os seus múltiplos. Os números restantes serão todos primos. Para construir este algoritmo pode-se inicializar todas as posições de um vetor de [2..N] posições com 1 (supõe-se que todos são primos), e a seguir para cada posição $P \leq \sqrt{N}$ do vetor contendo 1 coloca-se zero em todas as posições maiores e múltiplas de P. Fazer um algoritmo que determine e escreva todos os números primos até mil através do crivo de Erastóstenes. Abaixo um exemplo para determinar os primos até 27.

| | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| posição | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 |
| inicialização | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| múltiplos de 2 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| múltiplos de 3 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| múltiplos de 5 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

18. ☹ Fazer um algoritmo que lê números inteiros, armazenando-os em dois vetores, um vetor V com cem elementos e outro vetor W com dez elementos. Após verifique e escreva a primeira posição do vetor V onde toda a sequência de elementos do vetor W pode ser encontrada de forma contígua (todos os elementos juntos, exatamente um após o outro), caso contrário, escreva -1.

19. ☹ Fazer um algoritmo que leia vinte números reais armazenando-os em um vetor e a seguir:
- Identifique em que posição se encontra o menor elemento do vetor (ou um dos menores caso haja vários).
 - Troque o valor contido na posição identificada acima com o valor que está na primeira posição do vetor.
 - Identifique em que posição se encontra o menor elemento do vetor a partir da segunda posição do mesmo.
 - Troque o valor contido na posição identificada com o valor que está na segunda posição do vetor.
 - Identifique em que posição se encontra o menor elemento do vetor a partir da terceira posição do mesmo.
 - Troque o valor contido na posição identificada com o valor que está na terceira posição do vetor.
 - Coloque todos os demais elementos do vetor em ordem crescente desta forma (ordenação por seleção).
20. ☹ A ordenação pelo método da bolha (bubble sort), apesar de ter desempenho ruim, é fácil de implementar e funciona muito bem para uma quantidade de dados pequena. Este método funciona percorrendo todos os elementos (dados) várias vezes até que esses fiquem ordenados. Os elementos são percorridos de forma que cada elemento seja comparado com o subsequente e, caso não estejam na ordem desejada, são trocados. Fazer um algoritmo que leia cem números reais e escreva-os em ordem crescente utilizando o método da bolha.
21. ☹ Fazer um algoritmo que leia as notas finais dos trinta alunos de uma turma, calcule e escreva a média (μ) e o desvio padrão (σ) desta turma. O desvio padrão (σ) de uma população é igual a raiz quadrada da média dos quadrados das diferenças de cada valor (x_i) em relação a média (μ).
- $$\mu = \frac{1}{N} \sum_{i=1}^N x_i = \frac{x_1 + x_2 + x_3 + \dots + x_N}{N} \quad \sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2} = \sqrt{\frac{(x_1 - \mu)^2 + (x_2 - \mu)^2 + \dots + (x_N - \mu)^2}{N}}$$
- Uma outra forma de calcular o desvio padrão de uma população é:
- $$\sigma = \sqrt{\frac{1}{N} \left(\sum_{i=1}^N x_i^2 \right) - \mu^2}$$
- Qual a vantagem de usar esta forma? Como ficaria o algoritmo?
22. ☹ A mediana de uma população classificada (indivíduos ordenados), com uma quantidade ímpar de indivíduos é o elemento central e, para uma quantidade par de indivíduos a mediana é calculada como a média simples dos dois elementos centrais, ou seja, a mediana é um valor que divide essa população ao meio de forma que 50% dos valores são maiores ou iguais e 50% são menores ou iguais a mediana. Exemplos: Para os valores: 1 2 4 5 6, a mediana é 4 ; para os valores: 1 5 6 7 7 8, a mediana é 6,5. Fazer um algoritmo que leia números reais até que seja informado um número negativo ou no máximo duzentos números e escreva a mediana.
23. ☹ Fazer um algoritmo que leia números inteiros até que seja informado zero (descartando os negativos e o zero), ou no máximo quarenta números maiores que zero e escreva os números pares em ordem crescente e os ímpares em ordem decrescente.
24. ☹ Os vendedores de uma loja necessitam identificar rapidamente os preços a partir dos códigos dos produtos. Fazer um algoritmo que, inicialmente, leia e armazene os códigos e os preços dos produtos (máximo oitocentos produtos, ou até que seja informado código igual a zero) em dois vetores (um para os códigos e outro para os preços) e após permita consulta de preços a partir dos códigos. Durante o processo de consulta, para cada código informado, procurar a posição que o mesmo se encontra no vetor de códigos e escrevendo o preço da posição correspondente no vetor de preços ou uma mensagem apropriada caso o código não seja encontrado. Fazer um outro algoritmo que classifique os vetores de forma que os códigos fiquem em ordem crescente e após utilize pesquisa binária no processo de consulta dos produtos.
25. ☹ Fazer um algoritmo que leia no máximo quinhentos valores quaisquer (números reais) ou até que seja informado um número menor ou igual a zero (descartando-o) e escreva o menor valor único (menor valor que ocorre apenas uma vez). Desenvolva um outro algoritmo, mais eficiente, para o caso dos valores representarem uma quantidade monetária entre 0,01 e 50,00.
26. ☹ Fazer um algoritmo que leia oitocentos números inteiros. Escreva em ordem crescente somente os números que são únicos, ou seja, foram informados uma única vez. Exemplo: lê: 5 7 3 5 5 1 2 9 1 9 escreve: 2 3 7
27. ☹ Fazer um algoritmo que leia números inteiros até que seja informado um número negativo ou no máximo quarenta números e escreva os números que aparecem três ou mais vezes. Cada número que aparece três ou mais vezes somente pode ser escrito uma vez. Exemplo:

| | | | |
|-----|--|----------|-------|
| lê: | 10 12 30 20 10 12 12 25 12 25 20 59 20 67 80 11 -7 | escreve: | 12 20 |
|-----|--|----------|-------|

28. ☹ Fazer um algoritmo que leia o código e a nota de cem alunos e escreva somente os códigos em ordem decrescente de nota.

29. ③ No final do semestre, para efeito de registro, duas turmas, uma com dezessete e a outra com vinte alunos devem ser unidas em uma única turma com trinta e sete alunos. Para cada aluno há duas informações: código (real) e nota (inteiro). Os códigos dos alunos de cada turma estão classificados do menor para o maior. Fazer um algoritmo que leia o código e a nota dos alunos da turma A armazenando em dois vetores CA, NA; a seguir o código e a nota dos alunos da turma B armazenando nos vetores CB, NB (durante a leitura, para cada turma, cada novo código informado deve ser maior que o anterior, de forma que, os códigos fiquem em ordem crescente nos vetores CA e CB). Após o algoritmo deve juntar estes quatro vetores (CA, NA, CB, NB) em dois (COD, NOT) contendo os códigos e as notas de todos os alunos de forma que a classificação (ordem) por código seja mantida e finalmente escreva os códigos e as notas desta turma única. Exemplo para turmas A e B com sete e nove alunos respectivamente:

| | | | | | | | | | | | | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| CA | 3 | 4 | 7 | 9 | 15 | 18 | 22 | | CB | 2 | 6 | 12 | 14 | 17 | 19 | 20 | 21 | 24 |
| NA | 7,7 | 8,4 | 5,8 | 4,3 | 6,9 | 9,3 | 5,8 | | NB | 3,7 | 9,8 | 5,3 | 6,9 | 8,4 | 7,2 | 7,3 | 4,5 | 2,4 |
| COD | 2 | 3 | 4 | 6 | 7 | 9 | 12 | 14 | 15 | 17 | 18 | 19 | 20 | 21 | 22 | 24 | | |
| NOT | 3,7 | 7,7 | 8,4 | 9,8 | 5,8 | 4,3 | 5,3 | 6,9 | 6,9 | 8,4 | 9,3 | 7,2 | 7,3 | 4,5 | 5,8 | 2,4 | | |

30. ③ O consumo médio de combustível, que é mostrado a cada segundo no painel de um carro, é calculado a partir do consumo e da distância percorrida nos últimos trinta segundos. Fazer um algoritmo que leia (para facilitar, considerar que a leitura é efetuada exatamente a cada segundo), repetitivamente o odômetro (distância percorrida acumulada) [metros] e o consumo de combustível do último segundo [mililitros], calcule e escreva o consumo médio [quilômetros/litro] dos últimos trinta segundos e a velocidade média [quilômetros/hora] do último segundo. Arredondar a velocidade para o inteiro mais próximo e o consumo médio para o número com uma casa decimal mais próximo. O algoritmo deve finalizar quando for informado um número negativo para o consumo. Exemplo considerando a média dos últimos cinco segundos (em **negrito** alguns valores com respectivos cálculos demonstrados):

[illegible]

31. ☹ Fazer um algoritmo que leia vinte números reais e escreva quantas combinações três a três desses números (em posições distintas) que quando somados totalizem 100. Ex:

[illegible]

32. ☹ Fazer um algoritmo que leia novecentos números reais, determine e escreva a quantidade **mínima** de números necessários para que a soma desses seja maior ou igual a metade da soma total dos números lidos. Exemplo para dez números: 9,5 1,2 1,3 3,0 1,5 2,0 8,5 4,5 9,0 0,4 ; escreve: 3, pois bastam 3 números: 9,5 8,5 e 9,0 (soma=27,0) para atingir a metade da soma total ($40,9 / 2 = 20,45$).

33. ☹ Fazer um algoritmo que leia números até que seja informado zero (descartando-o) ou leia no máximo quarenta números armazenando-os em um vetor. Após o algoritmo deverá eliminar deste vetor os números negativos, deslocando os demais de forma que fiquem juntos no início do vetor, preservando a ordem original. Finalmente escreva os elementos restantes no vetor.

[illegible]

34. ☹ Fazer um algoritmo que leia uma quantidade indeterminada de números reais, positivos e/ou negativos, até que seja informado zero (descartando-o) e escreva os vinte maiores números lidos.
35. ☹ Os funcionários do departamento de compras de uma empresa desejam comparar a lista de preços do mês anterior com a lista de preços do mês atual de forma a identificar produtos excluídos, produtos incluídos e produtos com preços alterados. Cada lista contém vários códigos (inteiros) de produtos e respectivos preços (reais) e está classificada em ordem crescente de código. Fazer um algoritmo que leia os códigos e os preços destas duas listas escrevendo: os códigos e preços dos produtos que foram incluídos (não estão na lista anterior e estão na atual), os códigos dos produtos que foram excluídos (estão na lista anterior e não estão na atual) e os códigos e percentual de ajuste dos produtos que sofreram alteração de preço (estão nas duas listas, mas com preços diferentes). O algoritmo deverá ler no máximo duzentos produtos para cada lista ou até que seja informado zero no código e deverá garantir, durante a leitura de cada lista, que o código do produto que está sendo informado seja maior que o código anterior, caso contrário mostrar uma mensagem adequada e volta a ler o código até que isso ocorra. Exemplo:

| Lista Anterior | |
|----------------|-------|
| Código | Preço |
| 5 | 10,88 |
| 6 | 8,50 |
| 8 | 14,22 |
| 10 | 14,22 |
| 11 | 4,55 |
| 12 | 4,23 |
| 13 | 11,25 |
| 20 | 12,00 |

| Lista Atual | |
|-------------|-------|
| Código | Preço |
| 4 | 20,00 |
| 5 | 10,88 |
| 6 | 8,67 |
| 9 | 3,25 |
| 10 | 14,22 |
| 12 | 4,23 |
| 13 | 10,89 |

| O algoritmo deve escrever: |
|---------------------------------|
| Código 4: incluído preço 20,00 |
| Código 6: preço aumentou 2,00% |
| Código 8: excluído |
| Código 9: incluído preço 3,25 |
| Código 11: excluído |
| Código 13: preço diminuiu 3,20% |
| Código 20: excluído |