

1. Escreva um programa que faça o controle das informações relativas aos funcionários de uma determinada empresa. As informações que devem ser armazenadas são: Nome, Salário, Cargo e Idade. Defina a estrutura de dados e faça a leitura e exibição dos dados para **um** funcionário.
2. A loja VendeTudo possui 50 clientes. O gerente, para cada cliente, necessita armazenar o nome, cidade onde mora, o total da compra e o mês que o cliente comprou. Desenvolva o programa para ler e mostrar os dados.
3. A turma de Algoritmos e Programação B tem 17 alunos. Para cada aluno, é necessário armazenar o nome, número de faltas (em horas), os valores das nota 1, nota 2, nota 3 e a média final. A média das notas é a média aritmética e deve ser calculada. Após fazer a leitura dos valores, calcule a média das notas, para cada aluno e mostre o resultado final de acordo com as regras:
 - Para ser aprovado, o estudante deve ter 75% de frequência. Caso contrário, está Reprovado por Frequência.
 - Se o estudante tem 75% ou mais de frequência nas aulas, o resultado é:
 - Aprovado, se a média final for maior ou igual a 6,0 ou
 - Reprovado, se a média final for menor que 6,0

Para calcular a frequência, considere que a disciplina tem 80 horas.

4. Escreva um programa que leia um vetor de produtos, com 50 elementos, sendo cada elemento uma estrutura que contém o nome do produto, seu preço de compra e seu preço de venda. Depois de lido o vetor, retorne o nome e o índice do produto de menor preço de compra.
5. Escrever um programa que leia as coordenadas de três pontos cartesianos. Considerando que os dois primeiros pontos definem um retângulo de arestas paralelas aos eixos, verifique se o outro ponto está ou não dentro do retângulo.
6. Calcule a distância entre dois pontos no plano cartesiano, conforme a equação a seguir. Utilize uma estrutura para definir o tipo de dado.

$$d_{AB} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Nas questões 7, 8 e 9, faça a leitura dos dados como *string*, os converta para inteiros usando a função *atoi()* e use estruturas (*struct*) para armazená-los:

7. Fazer um programa que leia uma string no formato hh:mm:ss, converta os elementos desta strings em inteiros armazenando-os em uma estrutura e mostre o resultado na seguinte forma: “hh hora(s), mm minuto(s) e ss segundo(s).” Para esta questão, utilize a função *atoi()* que converte uma *string* em um número inteiro.
8. Escrever um programa que leia duas datas no formato dd/mm/aaaa e verifique qual dada ocorre primeiro. Use estruturas para armazenar os dados. Também é necessário o uso da função *atoi()* que converte uma *string* em um número inteiro.
9. Escrever um programa que lê um horário hh:mm:ss e verifica se este está correto.