1. Crie uma struct chamada ponto2d que tenha como atributos os pontos x,y e z. Crie duas estruturas do tipo ponto2d chamadas ponto\_incial e ponto\_final.

Faça um menu com as seguintes opções e implemente-as:

[1] – Digitar o valor do primeiro ponto

[2] – Digitar os valores do segundo ponto

[3] – Mostrar a distância entre os pontos

[4] – Sair

DICA: Distância entre dois pontos (x1, y1) (x2, y2): raiz quadrada de (x1 – x2)² + (y1 – y2)²

1. Crie uma struct para armazenar os dados (nome, idade, posição, numero\_camisa) dos jogadores de vôlei de um time. Crie duas estruturas para armazenar os jogadores principais e os reservas.

Desenvolva um programa em que o usuário consiga entrar com os jogadores principais (seis) e reservas (4) e mostre-os na tela. Para facilitar a entrada de dados, utilize um menu.

1. Faça um programa que declare 3 variáveis, dos tipos inteiro, real e caracter, atribua valores para estas variáveis e mostre seus valores assim como seu tamanho em bytes e seu endereço na memória (inteiro e hexa).
2. Crie um programa que declare duas variáveis (inteiro e real) atribua valores para elas e crie dois ponteiros, cada um apontando para uma destas variáveis.

Mostre na tela: o Valor e o endereço de memória das variáveis, o valor do ponteiro, o endereço dp ponteiro e o valor apontado pelo ponteiro.

Qual destes valores são iguais? Justifique?

1. Explique o que faz o programa a seguir, e quais valores serão mostrados ao final.

**int** x1 = 1, x2 = 2, \*px1;

px1 = &x1;

x1 = 5;

x2 = (\*px1) + 10;

x1 = 9;

printf(“x1: %d x2: %d \*px1: %d”, x1, x2, \*px1);

1. Modifique o exercício número 2 para a utilização de ponteiros na leitura e apresentação dos dados dos vetores.