





















































Questão 1.

2.5 P.

Um baralho tradicional possui 52 cartas descritas por um número ou nome e um naipe.

	1	2	3	4	5	6	7	8	9	10	Valete	Dama	Rei
Paus:													
Ouros:													
Copas:													
Espadas:													

1.0P Crie um tipo adequado para representar uma carta de baralho e em seguida inicialize um vetor de cartas com os valores possíveis de carta.

1.5P Crie uma função que recebe como parâmetro um vetor do tipo carta e embaralha aleatoriamente as posições das cartas no vetor. OBS: a função não imprime o baralho, use um loop na função **main** para realizar a impressão após a chamada da função.

Questão 2.

2.5 P.

1.0P Crie um tipo de dados para de reserva passagens de uma companhia aérea que contenha os seguintes dados:

- Número do voo
- Código do aeroporto de origem (três caracteres)
- Código do aeroporto de destino (três caracteres)
- Horário de partida (horas e minutos)

1.5P Agora, escreva um programa que leia os dados de 300 vôos e os exiba por ordem de partida crescente.

1. Baixe o arquivo voos.txt
2. Execute o comando abaixo no terminal

```
./a.out < voos.txt
```

OBS: se você criou o executável do seu programa com outro nome substitua o **a.out** pelo nome do seu executável.

Questão 3.

3 P.

Endereço IP é um endereço exclusivo que identifica um dispositivo na Internet ou em uma rede local.

O endereço IP é representado por um conjunto de quatro números: por exemplo, 192.158.1.38. Cada número do conjunto pode variar entre 0 e 255. Ou seja, o intervalo de endereçamento IP vai de 0.0.0.0 a 255.255.255.255.

1.0P Crie um tipo adequado para representar endereços IP.

1:0P Faça o tipo que você criou ocupar apenas 4 *bytes* de memória. Demonstre isso na função `main`.

1.0P Crie uma função para gerar inicializar variáveis do tipo IP aleatoriamente, exemplo: `randIP(x)` → `x = 154.34.2.250`. Use a função para inicializar um vetor de 30 posições do tipo IP e imprima o vetor na função `main`.

Questão 4.

2 P.

Baixe o arquivo `trem.c`. Complete o programa implementando a função `void printTrem(Vagao trem[], int tam)`.

Exemplo de saída

```
[20, primeira classe] [Trigo 5.90] [32, classe economica] [40, classe economica] [Granito 18.50]
```

A função só será considerada válida se levar em consideração o tipo de vagão para realizar a impressão de sua carga e for válida para qualquer tamanho de vetor do tipo `Vagao`.