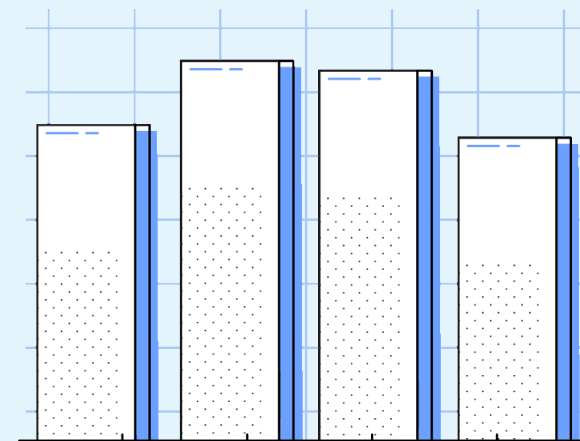


Estrutura de Dados / conceito (struct)

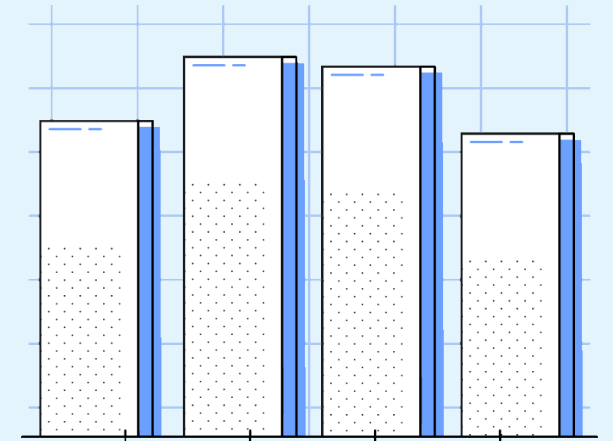
Como todos sabemos, nem todos os dados que precisamos usar podem ser agrupados em matrizes. Frequentemente usamos dados de diversos tipos diferentes, com tamanhos diferentes. Para tipos de dados de diferentes tamanhos existem estruturas de armazenamento de dados heterogêneos.

O especificador **struct** é usado para esta finalidade. Com ele podemos criar tipos que armazenam dados compostos por agrupamentos de outros tipos primitivos da linguagem. Geralmente, os dados são armazenados de forma a facilitar a identificação de cada campo de dados que pretende-se manter, para isso usamos nomes para cada campo dentro da estrutura de dados, de forma a ter um meio de acessá-la depois.



Estrutura de Dados / conceito (struct)

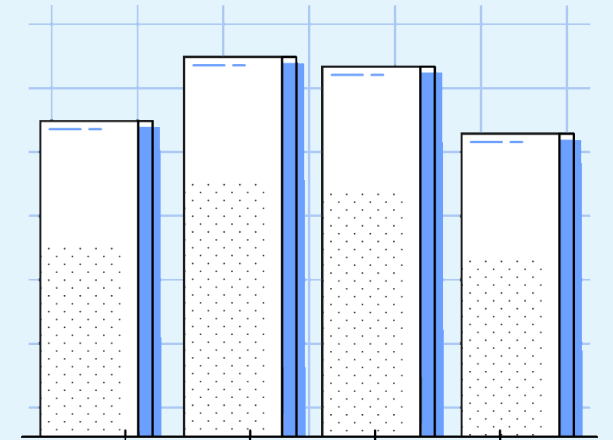
Estruturas são blocos básicos de informação e são manipulados de maneira primitiva. Basicamente o compilador instrui a montagem de um código que manipula-as de forma a copiar, referenciar e obter posição na memória. Todas as outras formas de tratar os dados devem ser providas pelo código do programa.



Estrutura de Dados / conceito (struct) Implementação

Para criar um tipo de dados composto heterogêneo, basicamente, cria-se uma lista de tipos e nomes de variáveis separadas por ponto e vírgula. Podemos imaginar esta lista como um bloco de código, em linguagem C, onde estão presentes apenas as declarações de variáveis. Para isso temos a seguinte sintaxe:

```
struct Estrutura
{
    <Tipo A> NomeVariavelA;
    <Tipo A> NomeVariavelA2;
    <Tipo A> NomeVariavelA3;
    <Tipo B> NomeVariavelB;
    <Tipo C> NomeVariavelC;
    <Tipo D> NomeVariavelD;
    ...
    ...
    <Tipo Z> NomeVariavelZ;
} [<NomeVariavelComposta>];
```



Estrutura de Dados / conceito (struct) Implementação

1º

struct Locadora

```
{  
    char titulo[30];  
    char genero[25];  
    int duracao[30];  
    float valor[30];  
  
};
```

INT MAIN(){

STRUCT LOCADORA ALUGAR1, ALUGAR2,ALUGAR3;

}

2º

struct Locadora

```
{  
    char titulo[30];  
    char genero[25];  
    int duracao[30];  
    float valor[30];  
  
}; ALUGAR1, ALUGAR2,ALUGAR3;
```