

Atividade de Contêiner Vector

Crie o projeto ProjetoCarroMenu a) Neste projeto faça: i) Crie a classe Carro que tenha como atributos cor, ano e modelo. Nesta classe crie outro construtor que recebe como parâmetro os dados do objeto e os métodos getters e setters. ii) Crie o arquivo que conterà o método main. Nele crie um vetor de objetos da classe Carro utilizando o contêiner vector e a partir do menu a seguir efetue os devidos cálculos:

Menu

" 0 - Sair "

" 1 - Inserir "

" 2 - Imprimir "

" 3 – Qtde de Carros da cor Prata"

```
1  #include <string>
2  #include <sstream>
3
4  using namespace std;
5
6  class CarroMenu{
7
8      private:
9
10     string cor;
11     int ano;
12     string modelo;
13
14     public:
15
16     CarroMenu();
17     CarroMenu(string cor, int ano, string modelo);
18     ~CarroMenu();
19     void setCor(string cor);
20     void setAno(int ano);
21     void setModelo(string modelo);
22     string getCor();
23     int getAno();
24     string getModelo();
25     string toString();
26
27 };
28
```

Atividade de Contêiner Vector

```
1  #include "CarroMenu.h"
2  CarroMenu::CarroMenu()
3  {
4  }
5
6  CarroMenu::CarroMenu(string cor, int ano, string modelo){
7      this->cor = cor;
8      this->ano = ano;
9      this->modelo = modelo;
10 }
11
12 CarroMenu::~CarroMenu()
13 {
14 }
15
16 void CarroMenu::setCor(string cor){
17     this->cor = cor;
18 }
19 void CarroMenu::setAno(int ano){
20     this->ano = ano;
21 }
22 void CarroMenu::setModelo(string modelo){
23     this->modelo = modelo;
24 }
25 string CarroMenu::getCor(){
26     return cor;
27 }
28
29 int CarroMenu::getAno(){
30     return ano;
31 }
32
33 string CarroMenu::getModelo(){
34     return modelo;
35 }
36
37 string CarroMenu::toString(){
38     string str_ano;
39     stringstream ss_ano;
40
41     ss_ano << ano;
42     str_ano = ss_ano.str();
43
44     return "[CarroMenu] cor:" + cor + " Ano:" + str_ano + " modelo: " + modelo;
45
46
47 }
```

Atividade de Contêiner Vector

```
1  #include <iostream>
2  #include <string>
3  #include <vector>
4  #include "CarroMenu.h"
5  using namespace std;
6
7  int menu() {
8      int op;
9      do{
10         cout << "Menu de Opções" << endl;
11         cout << "0 - Sair" << endl;
12         cout << "1 - Inserir" << endl;
13         cout << "2 - Imprimir" << endl;
14         cout << "3 - Qtde de Carros da cor Prata" << endl;
15
16
17         cin >> op;
18     } while (op<0 || op>3);
19     return op;
20 }
21
22 CarroMenu preencher() {
23
24     string auxCor;
25     int auxAno;
26     string auxModelo;
27
28     cin.ignore(); // buffer Limpa a memoria
29     cout << "Digite a cor:" << endl;
30     cin >> auxCor;
31     cout << "Digite o ano:" << endl;
32     cin >> auxAno;
33     cout << "Digite o modelo:" << endl;
```

Atividade de Contêiner Vector

```
34     cin >> auxModelo;
35     return CarroMenu(auxCor, auxAno, auxModelo);
36 }
37 void imprimir(vector<CarroMenu> vet){
38     for (int i = 0; i <vet.size(); i++){
39         cout << vet[i].toString() << endl;
40     }
41 }
42 // CASO 3
43
44 void calcoQtdePrata(vector<CarroMenu>vet){
45     string cor;
46     int qtdep;
47     vector<CarroMenu>::iterator p;
48     for (p=vet.begin(); p!=vet.end(); p++){
49         if ((*p).getCor() == Prata){
50             qtdep = qtdep + 1;
51         }
52     }
53
54     cout << "A quantidade de carros prata é: " << qtdep << endl;
55 }
56 }
57 }
58 int main(){
59     setlocale(LC_ALL, "Portuguese");
60     vector<CarroMenu> vetor;
61     int qtde = 0, opcao;
62     CarroMenu novo;
63 }
```

Atividade de Container Vector

```
64 do{
65     opcao = menu();
66     switch (opcao){
67
68
69     case 1:
70         if (vetor.size() == 0){
71             cout << "não há dados para serem impressos" << endl;
72         }
73         else{
74             novo = preencher();
75             vetor.push_back(novo);
76         }
77
78         break;
79     case 2:
80         if (vetor.size() == 0){
81             cout << "não há dados para serem impressos" << endl;
82         }
83         else{
84             imprimir(vetor);
85         }
86         break;
87     case 3:
88         if (vetor.size() == 0){
89             cout << "não há dados para serem impressos" << endl;
90         }
91         else{
92             calcQtdePrata(vetor);
93         }
94         break;
95
96     }
97
98 }while (opcao != 0);
99
100 }
```