

Programação Orientada a Objetos

Organização do código apresentado

1.Quais foram as classes consideradas na primeira versão da aplicação que foi testada?

- Carro
 - criar lo
 - obter um piloto
 - gasto de energia
- Piloto
 - criar lo
 - entrar no carro

2.Quais os conceitos/classe que identificou ao ler o enunciado?

- carro
- piloto
- dgv
- geral
- autodromo

3.Relativamente a duas das principais classes da aplicação, identifique em que classes ou partes do programa são criados, armazenados e destruídos os seus objectos.

Carro: os objectos desta classe são criados, armazenados e destruídos na classe dgv.
Piloto: os objectos desta classe são criados, armazenados e destruídos na classe dgv.
dgv: os objectos desta classe são criados, armazenados e destruídos na classe geral.
atd: os objectos desta classe são criados, armazenados e destruídos na classe geral.

4. Indique um exemplo de uma responsabilidade atribuída a uma classe que esteja de acordo com a orientação dada acerca de Encapsulamento.

O dgv é responsável pelos carros e pilotos.

5. De entre as classes que fez, escolha duas e justifique por que considera que são classes com objetivo focado, coeso e sem dispersão.

Classe Carro: tem dados e responsabilidades relativos apenas aos carros, como a sua velocidade, consumos e características.
Classe Piloto: tem dados e responsabilidades relativos apenas aos pilotos, como o seu nome e em que carro está sentado.

6. Relativamente à aplicação entregue, quais as classes que considera com responsabilidades de interface com o utilizador e quais as que representam a lógica?

Todas elas tem responsabilidades de logica e verificacao.

Quem tem responsabilidade da interface com o utilizador é a classe geral.

7. Identifique o primeiro objecto para além da camada de interacção com o utilizador que recebe e coordena uma funcionalidade de natureza lógica?

As ordens vindas da camada de interacção com o utilizador são recebidas e processadas por um objeto da classe geral..

8. A classe que representa a envolvente de toda a lógica executa em pormenor muitas funcionalidades, ou delega noutras classes? Indique um exemplo em que esta classe delega uma funcionalidade noutra classe.