

## Lista de revisão 3

**Questão 1)** Explique com suas palavras e dê exemplos em C++ dos seguintes conceitos:

1. Classe
2. Objetos
3. Atributos
4. Métodos
5. Encapsulamento
6. Herança
7. Polimorfismo

**Questão 2)** Escreva um programa em C++ que defina uma classe chamada *FormaGeometrica* com um construtor que **receba** e dê valor aos atributos **largura** e **altura**. Defina duas subclasses triângulo e retângulo, que possuem o método *CalculaArea()*. Na *main*, defina um objeto triângulo e um retângulo, e então chame o método *area()* a partir destes dois objetos.

**Questão 3)** Crie uma classe em C++ chamada **InteiroGigante** que usa um array de dígitos de 40 elementos para armazenar inteiros tão grandes quanto 40 dígitos. Forneça os métodos **leiaInteiroGigante**, **imprimeInteiroGigante**, **adicioneInteiroGigantes** e **subtraiaInteiroGigantes**. Para comparar objetos do tipo *InteiroGigante*, providencie métodos **eIgual**, **naoEIgual**, **ehMaior**, **ehMenorQueOuIgualA** e **ehMenorQueOuIgualA** – cada um desses métodos simplesmente retorna *true* se a relação entre os dois inteiros gigantes for verdadeira e retorna *false* se a relação não for verdadeira.

**Questão 4)** Explique a diferença de um erro e uma exceção, e explique quando devemos lançar e tratar uma exceção.

**Questão 5)** Dado o código a seguir, diga o que será mostrado na tela.

```
#include <iostream>
#include <string>
using namespace std;
class Pai{
public:
    Pai(){
        cout<<"\nSubclasse, eu sou seu pai.";
    }
    Pai(string nome){
        cout<<"\n"<<nome<<"", eu sou seu pai."
        ;
    }
    void Falar(){
        cout<<"\nOla Mundo";
    }
};
class Filho : public Pai{
public:
    Filho(){
        cout<<"\nNAAAAAO";
    }
    void Falar(){
        cout<<"\nAte mais mundo";
    }
};
int main() {
    Pai x,y("Jorge");
    Filho a;
    a.Falar();
    x.Falar();
}
```

**Questão 6)** Informe e explique os possíveis erros que podem ocorrer no código abaixo:

```
#include <iostream>
using namespace std;
class Agente{
public:
    float dinheiro;
    int pontos;
};
class Raze : Agente{
    void Falar(){
        cout<<"\nBombinho ja foi!";
    }
public:
    int numeroDeBombinhas;
};
int main() {
    Agente a1;
```

```
Raze r1;  
r1.dinheiro = 100;  
r1.Falar();  
r1.numeroDeBombinhas = 2;  
a1.numeroDeBombinhas = 1;  
return 0;  
}
```