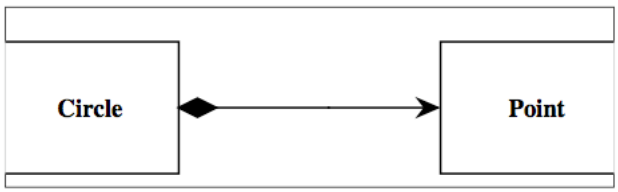
Nome: _____ Matrícula: _____

1. A composição é um tipo de associação que possui uma semântica muito própria. Por exemplo, na figura abaixo estamos dizendo que a classe `Point` é parte da classe `Circle`, e, mais ainda, instâncias de `Point`, nesta composição, somente existem enquanto o objeto possuidor, instância de `Circle`, ainda existir.



Em algumas linguagem, como C++, é fácil escrever relações de composição, basta que um objeto seja declarado como parte do escopo local de uma classe, por exemplo:

```
#include <iostream>
class Obj1 {
  public:
    Obj1() {std::cout << "Creating obj1\n";}
    ~Obj1() {std::cout << "Deleting obj1\n";}
    int p;
};
class Obj2 {
  public:
    Obj2() {std::cout << "Creating obj2\n";}
    ~Obj2() {std::cout << "Deleting obj2\n";}
    Obj1 o;
};
int main() {
  Obj2 *o = new Obj2();
  delete o;
}
```

Neste caso, quando a instância de `Obj2` criada no método `main` for desalocada, então a instância de `Obj1` também o será. Por outro lado, em Java não existe a possibilidade de um objeto possuir outro por valor. Tudo são referências. Como, então, sabemos que um objeto possui outro por composição?