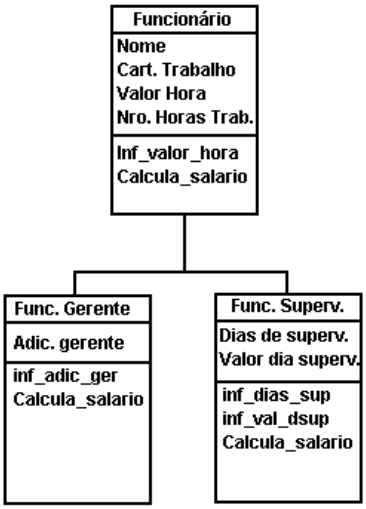
Classes em Detalhes

Prof. Leandro Buss Becker
mailto:lbecker@inf.pucrs.br

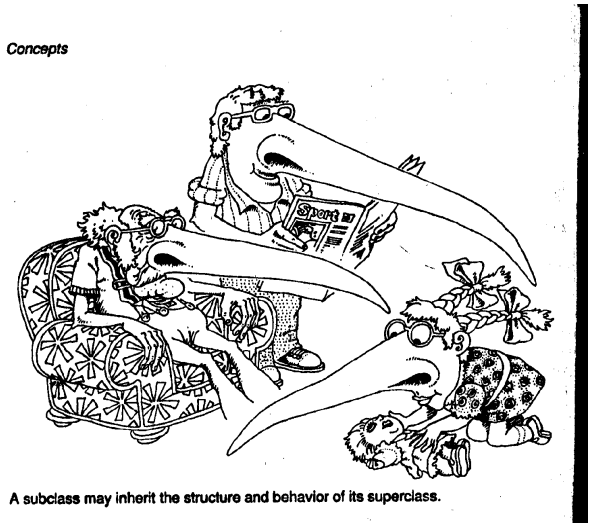
Herança

- possibilita a derivacão de tipos mais específicos a partir de um tipo mais genérico
- sub-classes herdam os dados e os métodos da super-classe, permitindo ainda modificações nos mesmos
- em geral, herança é visto como uma forma de hierarquia de tipos (sub-classes pertencem ao tipo das super-classes)
- Herança simples: uma sub-classe deriva de somente uma super-classe
- Herança múltipla: uma sub-classe deriva de várias super-classes

Herança

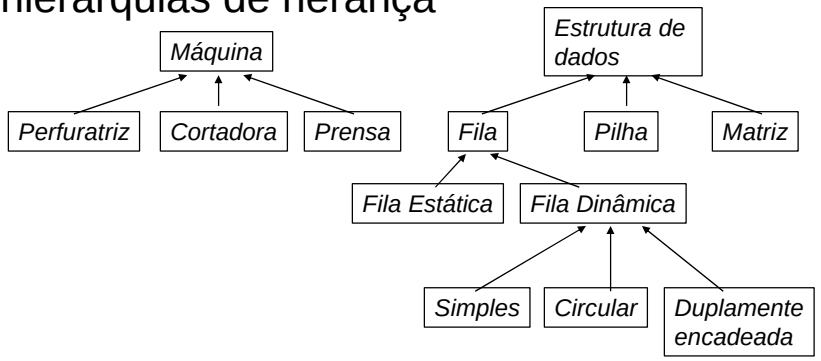


Herança



HERANÇA

- Classes podem ser organizadas em hierarquias de herança



Herança

- uma subclasse pode redefinir métodos herdados da superclasse, modificando uma parte do seu comportamento
- especialização vs. generalização

Herança

- classes podem dar origem a novas classes, como forma de estender ou modificar um determinado comportamento
- o mecanismo de herança permite que uma nova classe seja gerada com base em classes já existentes

Herança

```
class A {
 private:
    int Apri, AFpriv();
 protected:
    int Apro, AFpro();
 public:
    int Apub, AFpub();
}
```

```
class B : public A {
 private:
    int Bpri, BFpriv();
 protected:
    int Bpro, BFpro();
 public:
    int Bpub, BFpub();
}
```

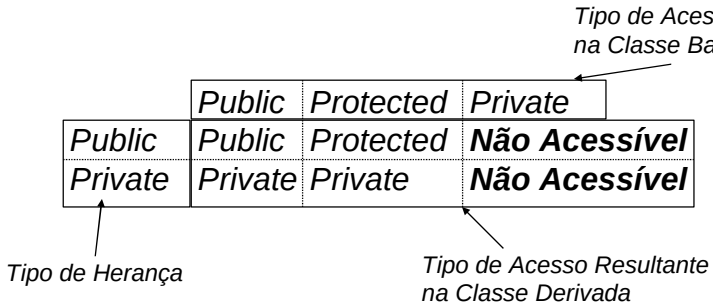
```
class C : private A {
 private:
    int Cpri, CFpriv();
 protected:
    int Cpro, CFpro();
 public:
    int Cpub, CFpub();
}
```

Tanto B como C tem direito de acesso à:
Apro,Apub, AFpro(),AFpub()

Externamente, B da acesso à:
Bpub,BFpub(),Apub, AFpub()

Externamente, C da acesso à:
Cpub,CFpub()

Regras de Visibilidade



Construtores Parametrizados e Herança

```
class C1  {
      int x,y;
  public:
    C1(int px,int py) { x = px; y = py;  }
  };
class C2 : public C1  {
    int b,c,d;
  public:
    C2(int a);
  };
C2::C2(int a) :C1(a,a*2)  {
    b = a;   c = a*2;  d = 23;
  }
```

```
void main() {
   C2 n(12);
  }
```

Referências Bibliográficas

- Schield, H. “*Expert C++*”.
- Pereira et al. Apostila de C++.