

## 5.3 Exercícios

5.1 Um programador C++ codificou o seguinte módulo como parte de sua aplicação:

---

```
int a;  
int umaFuncao() { return a; }  
void umaFuncao(int v) { a = v; }  
int umaFuncao(int v, int b) { a = v + b; return a; }
```

---

Após compilar este código com o compilador g++, ele analisou o código objeto associado com o aplicativo nm do sistema Unix, que apresenta a tabela de símbolos do módulo objeto, e encontrou as seguintes definições para os nomes das funções:

```
0000000a T _Z9umaFuncao  
00000018 T _Z9umaFuncaoii  
00000000 T _Z9umaFuncaov
```

- (a) Explique, a partir desse exemplo, como esse compilador faz a decoração de nomes de funções.
- (b) Antes de chegar ao programa anterior, o programador havia tentado compilar, sem sucesso, a seguinte versão de seu código, na qual também utilizava a sobrecarga de nomes de funções:

---

```
int a;  
int umaFuncao() { return a; }  
void umaFuncao(int v) { a = v; }  
    int umaFuncao(int v) { a = v; return a; }
```

---

Explique qual foi a causa do problema que o programador encontrou nessa compilação.

5.2 Um programa em C++ contém as seguintes declarações de variáveis:

```
char c = '0';  
int i = 1;  
float f = 2.0;
```

Quais dos seguintes comandos causariam erros detectados na etapa de análise semântica e, nesse caso, qual tipo de verificação captura o erro?

- (a) `c = i;`
- (b) `i = c;`
- (c) `c % i;`
- (d) `c « i;`
- (e) `f « i;`
- (f) `f % c;`