

```
//Preparar declarações iniciais
```

```
void S()
{
    switch (tok)
    {
        case BEGIN      : eat (BEGIN); S(); L(); break;
        case IF          : eat (IF); E(); eat (THEN); S(); eat (ELSE); S(); break;
        case PRINT       : eat (PRINT); eat(ID); break;
        default: ERRO();
    }
}
```

```
void L()
{
    switch (tok)
    {
        case ;          : eat (;); S(); L(); break;
        case END        : eat (END); break;
        default: ERRO();
    }
}
```

```
void E()
{
    eat (ID); eat(=); eat(NUM);
}
```

```
void eat(int t)
{
    if (tok == t)
        advance();
    else ERRO();
}
```

```
void advance()
{
    tok= getToken();
}
```

```
int getToken()
{
    // chama o analisador léxico e obtém
    // o próximo token do programa fonte
}
```

```
void ERRO()
{
    printf("ERRO na Análise Sintática: %d %d", line, tok);
}
```

```
int main()
{
    advance()
    S();
    return 0;
}
```