

O artigo tem o objetivo de explicar dentro das linguagens de programação os tipos, a abstração de dados e o polimorfismo. Herança de tipos, abstração são muito bem explicadas por linguagens orientada por objetos. Ele também abrange o monomorfismo e polimorfismo.

Ele começa o artigo citando que no passado existia apenas um universo, o não-tipado, ou melhor, com apenas um tipo. E logo após ele abre falando dos motivos de se precisar de tipos em linguagens de programação. Esses motivos citados por ele equivalem a organização que fica muito mais fácil quando existem diferentes tipos.

Um tipo pode servir como uma armadura, explica Cardelli, pois na tipagem forte e estática ao qual ele explica, ele fala que nesse sistema de tipos objetos ficam protegidos e limita como os objetos podem interagir no programa, e sem essa tipagem os objetos ficam liberados para qualquer um ver e interagir.

Cardelli fala que a tipagem forte e estática é uma propriedade poderosa, porém todas as variáveis e expressões ficam limitadas a um a um tempo de compilação as vezes muito restritivas.

Cardelli também cita os tipos estático e forte em seu artigo, falando a utilidade de cada um dos dois. A linguagem de programação estáticamente tipada são as que as expressões do programa são determinadas por uma análise estática de um programa, ou seja, ela possibilita que a inconsistência de tipos sejam descobertas ao compilar um programa, melhor explicando, ela garante a consistência de tipos.