

Exploring Polymorphic Type System

FA17-IN-CSCI-56500-20532

Programming Language

Project Report

Under the guidance of: Dr. Rajeev R. Raje

By

Mani Manjusha Kottala

Yashwanth Reddy Kuruganti

Aakarsh Nadella

Abstract

A Type system ensures the property of maintaining data types associated with constructs of a program, such as functions, variables etc., A type is a property that characterizes the construct to determine what data is exactly stored or what value will be returned by that construct. These types help a program to behave in a consistent manner thus minimizing the chances of errors or bugs. In this project, we discuss polymorphism concepts from the perspective of type systems, understand and implement in both static and dynamic approaches and compare their mechanism and outputs in two programming languages, Ada and Python.

Table of Contents

1.	Introduction	1
2.	Problem Statement	1
3.	Polymorphism	1
4.	Polymorphism in ADA	
4.1	Parametric polymorphism	2
4.2	Inclusion Polymorphism	2
4.3	Method Overloading	3
4.4	Operator Overloading	3
4.5	Coercion	3
5.	Polymorphism in Python	
5.1	Parametric Polymorphism	3
5.2	Method Overriding	4
5.3	Duck Typing	4
5.4	Function Overloading	
5.4.1	*args (magic variables)	4
5.4.2	Optional Parameters	5
5.4.3	Single dispatch	5
5.5	Operator Overloading	5
5.6	Type Coercion	6
6.	Comparison of Ada and Python	
6.1	Based on Syntax	6
6.2	Polymorphic Factor	6
6.3	Comparison of Polymorphism	6
7.	Conclusion	7
8.	Critique	7
	References	7