

# **Design and Implementation of an Algol-Like Intermediate Compiler.**

Ajila, Samuel Adesoye

M.Sc. Computer Science

Department of Computer Science

Obafemi Awolowo University, Ile Ife, Nigeria

1987.

## **Abstract:**

The gap created by the abstraction of a programming language from the machine level is bridged by means of a translation system known as a compiler. The design and implementation of this intermediate compiler is based on the Recursive Descent compiling technique. In this technique, the compiler is activated through the syntax analyser.

The syntax analyser is divided into a number of recognition routines, each of which has the task of checking whether a particular kind of phrase is present in the input, and then calls upon the services of other routines to recognize the appearance of other subphrases. The routines are mutually recursive. In addition, semantic checking and code generation phases are integrated into the syntax analysis phase.

Some of the results of this technique include the ability of the intermediate compiler to do its recognition, type checking and code generation without 'backup'. The total code of the compiler is very small and it is written in a high-level programming language (PASCAL).

In general these advantages result in portable and more easily understood compiler. As a result of the implementation technique, this compiler can be used as a teaching aid in compiler construction class at the undergraduate level. It can also be used to introduce structured programming for a beginner in Computer Science.

**Keywords:** Programming language/ compiler/ Recursive Descent compiling technique/ syntax analyser

**Supervisor:** G. O. Owoso