



UNIVERSIDAD PANAMERICANA

Compilers
COMPUTER GRAPHICS AND SYSTEMS ENGINEERING

PL/0 and PL/1

Semester 2022-2

Student:

(0197495@up.edu.mx) Mariana Ávalos Arce

Professor:

Blanca Estela Villalvazo Flores

Monday, May 18th, 2022
Guadalajara, Jalisco

1 Introduction

The following document contains a brief explanation of what is PL/0 and PL/1 languages, as well as a bit of a glimpse to the life and achievements of Niklaus Wirth, who had a strong relationship with the mentioned languages. At the end of the document some personal conclusions are presented.

2 Who is Niklaus Wirth?

Niklaus Wirth is an Electrical Engineer and Doctor in Computer Science who was born in Switzerland, in 1934. His outstanding work in the field of Computer Science earned him an ACM Turing Award in 1984.

Wirth began his pioneering work in computer science by creating two languages: Euler (based on his dissertation work), and the systems programming language PL360 (for the IBM System/360 series of computers). He made new discoveries in the branches of **syntax and semantics of compilers**, in novel implementation techniques, and in careful language design for efficient implementation with specific parsing methods, as well as **pioneering the formal description of programming language syntax**.

He was part of the creation of many programming languages, such as Algol (the company for which he worked) EULER, FORTRAN, Pascal, PL360 (for IBM) and **PL/0 and PL/1**. The latter will be described in the next section. The languages in which Wirth worked in were ground-breaking for the hardware of his time, including **recursive functions, structured blocks, and the concept of scope in variables**.

By working in Algol, Wirth collaborated with other future Turing Award winners including C.A.R. (Tony) Hoare, **Edsger Dijkstra**, and Peter Naur. Like Wirth, all had joined the group after showing early promise as designers of compilers and other systems software. But perhaps Wirth's most influential creation was Pascal Programming Language in 1969: was simple, flexible and designed for rapid compilation into efficient code. Wirth is today one of the most important figures regarding Compiler Theory. Here is one of his thoughts:

...every single project was primarily a learning experiment. One learns best by inventing. Only by actually doing a development project can I gain enough familiarity with the intrinsic difficulties and enough confidence that the inherent details can be mastered.

3 PL/0 and PL/1

3.1 PL/0

PL/0 is a programming language introduced by Niklaus Wirth, especially **designed for educational purposes**. PL/0 is very similar to Pascal, also written by Wirth, but the difference is that PL/0 was much simpler, since it was presented in Wirth's book *Algorithms + Data Structures = Programs* in 1976. Wirth created PL/0 as an explanation of how to construct a compiler. It is a general purpose language, but it is quite limited due to its purposes: it does not include real numbers, it features very basic arithmetics and only supports **if** and **while** statements.

The Syntax rules for PL/0 Language are shown below in Extended Backus-Naur Form:

```
program = block "." ;

block = [ "const" ident "=" number {" "," ident "=" number} ";" ]
       [ "var" ident {" "," ident"} ";" ]
       { "procedure" ident ";" block ";" } statement ;
```

```

statement = [ ident ":" expression | "call" ident
              | "?" ident | "!" expression
              | "begin" statement { ";" statement } "end"
              | "if" condition "then" statement
              | "while" condition "do" statement ];

condition = "odd" expression |
            expression ("=" | "<" | "<=" | ">" | ">=") expression ;

expression = [ "+" | "-" ] term { ("+" | "-") term };

term = factor { ("*" | "/" ) factor };

factor = ident | number | "(" expression ")";

```

His later book *Compiler Construction* (1976) provided the full source code of PL/0 compiler written in Pascal.

3.2 PL/1

PL/1 is a programming language proposed and published by IBM in 1970, where Wirth also had the opportunity to contribute to. Unlike PL/0, this language is **designed for academic, commercial and industrial use**. It is still used as of today. PL/I is a block-structured language, consisting of packages, procedures, statements, expressions, and built-in functions.

A **block** can be either a subroutine, or just a group of statements. A PL/I block allows you to produce highly modular applications, because blocks can contain declarations that define variable names and storage classes. Therefore, this language introduced Wirth's innovation of **variable scopes**.

A PL/I block is either a PROCEDURE or a begin block, any of which contains zero or more statements and/or zero or more blocks. A procedure is a sequence of statements. Below is the structure of a Procedure Block and a Begin Block:

Procedure Block

```

A:  procedure;
      statement-1
      statement-2
      .
      .
      .
      statement-n
end Name;

```

Begin Block

```

B:  begin;
      statement-1
      statement-2
      .
      .
      statement-n
end B;

```

4 Conclusions

One of the main conclusions for me personally is that for a person to be called the creator of a programming language, that person not only creates a new syntax for commands, but also the compiler and grammar rules that make that language stand alone and be able to achieve its purpose: help people solve problems. Also, a programming language is an interface for people to communicate with machines, that is, make machines perform computations needed for academic or industry problems.

References

- [1] IBM Corporation. *PL/I program structure*. URL: <https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-pli-program-structure>.
- [2] Thomas Haigh. *NIKLAUS E. WIRTH*. URL: https://amturing.acm.org/award_winners/wirth_1025774.cfm.
- [3] Hans Otten. *PL/0*. URL: <http://pascal.hansotten.com/niklaus-wirth/pl0/>.
- [4] Progopedia. *PL/0 Programming Language*. URL: <http://progopedia.com/language/pl0/>.