

## APÊNDICE - Caio César Pereira

Pascal é uma linguagem de programação com grande relevância histórica mas que perdeu parte de sua popularidade e uso ao longo do tempo. Isso se torna ainda mais evidente ao compará-la com outras linguagens que surgiram em períodos semelhantes, como C em 1972 ou FORTRAN em 1957, mas ainda são amplamente utilizadas. Embora tenha sido criada há muitos anos, Pascal compartilha diversas características com linguagens modernas. Abaixo estão alguns exemplos denotando suas semelhanças e diferenças.

Exemplo de função na linguagem de programação "Go":

```
func sum(x int, y int) int {  
    return x + y  
}
```

Exemplo de função na linguagem de programação "Pascal":

```
function sum(x, y: integer): integer;  
begin  
    sum := x + y;  
end;
```

Nos exemplos acima, percebe-se que Pascal não possui o comando "return" (amplamente utilizado em linguagens modernas), em vez disso, cada função tem uma variável interna com o mesmo nome, à qual o valor de retorno é atribuído. Outra diferença marcante está na forma como os blocos de código são delimitados: enquanto Pascal utiliza "begin" e "end", Go emprega "{" e "}". Essa abordagem também se reflete em operadores como "and", "or" e "not", que em Go correspondem a "&&", "||" e "!", respectivamente. Tais características são parte da herança de linguagens da família ALGOL, que priorizam palavras-chave em vez de símbolos para operadores lógicos e delimitadores.

Outro aspecto marcante da linguagem é sua capacidade de manipular o fluxo de execução, algo que em linguagens modernas é fortemente desencorajado. Para isso, Pascal faz uso de "labels" e do comando "goto", que permite redirecionar a execução do programa para um ponto específico no código. Quando o comando "goto" é utilizado, a execução salta diretamente para o label referenciado, como ilustrado no exemplo abaixo ao apontar para o label "1".

```
program exemplo;
label 1;
var
    a, b : integer;

begin
    a := 10;
    b := 0;
    repeat
        if(a = 15) then begin
            b := a * 2;
            goto 1;
        end;

        a := a + 1;
    until a = 20;
    writeln('Valor final de a:', a); // não será executada se o fluxo de execução for
alterado
    1: writeln('Valor final de b: ', b); // label 1
end.
```

Em conclusão, apesar de não estar mais entre as linguagens mais utilizadas, Pascal introduziu muitos conceitos e práticas que foram além de seu tempo para serem implementadas, ou desencorajadas, em linguagens modernas e ainda hoje mantém seu valor educacional e histórico.