

# Algoritmos

## Introdução aos Algoritmos

Prof. Dr. Osmar Betazzi Dordal

1

# Introdução aos Algoritmos

# Breve História

- A origem da computação como a conhecemos remete aos primórdios da sociedade humana, pois o **homem** sempre teve a **necessidade de construir equipamentos e máquinas que realizassem cálculos**.
- A primeira máquina construída com o objetivo de auxiliar nas operações matemáticas surgiu por volta de **3500 a.C.**, na região da Mesopotâmia (vale dos rios Tigre e Eufrates, atual Iraque) e foi batizada de **ábaco** (ALCALDE; GARCIA; PENUELAS, 1991).

## Breve Histórico

O **Ábaco** influenciou muitas outras culturas, como da Babilônia, Egito, Grécia, Roma, Índia, China e Japão durante vários séculos.





## Breve Histórico

- Somente em 1642, o francês Blaise Pascal cria uma máquina com capacidade para realizar **somas e subtrações** de forma **automática**.
- **Pascalina: Simulava** por meio de rodas dentadas o funcionamento do **Ábaco**



Máquina de somar e subtrair de Pascal

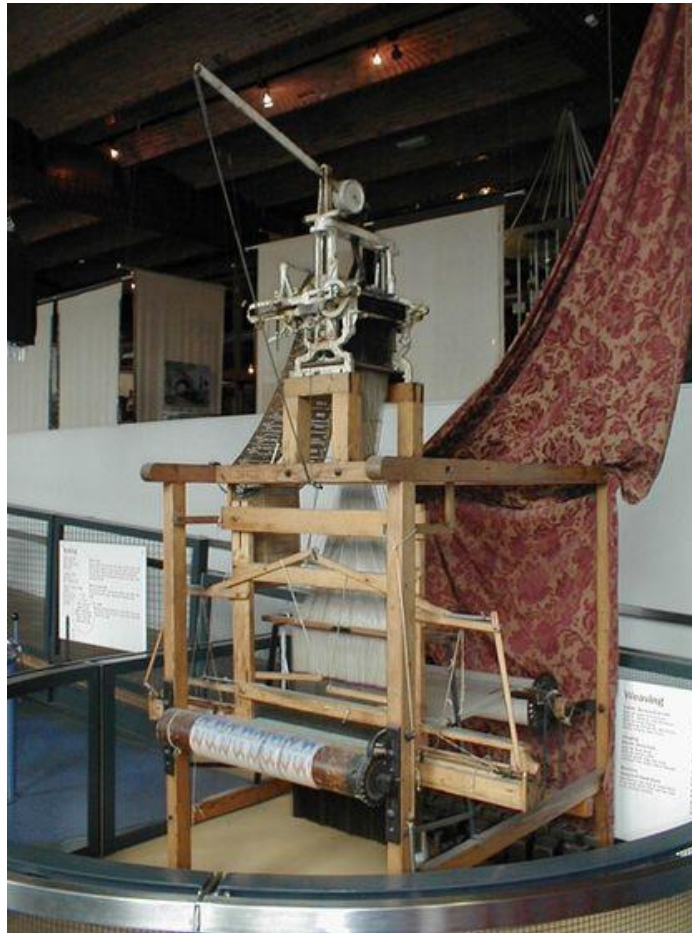
# Breve História

- Diversas máquinas ao longo do tempo foram criadas.
- O objetivo sempre foi auxiliar em cálculos complexos ou substituir o homem em algumas atividades
- Entre outras máquinas foram criadas:



## Breve Histórico

- Máquina de tear (1804), de Joseph Marie Jacquard.

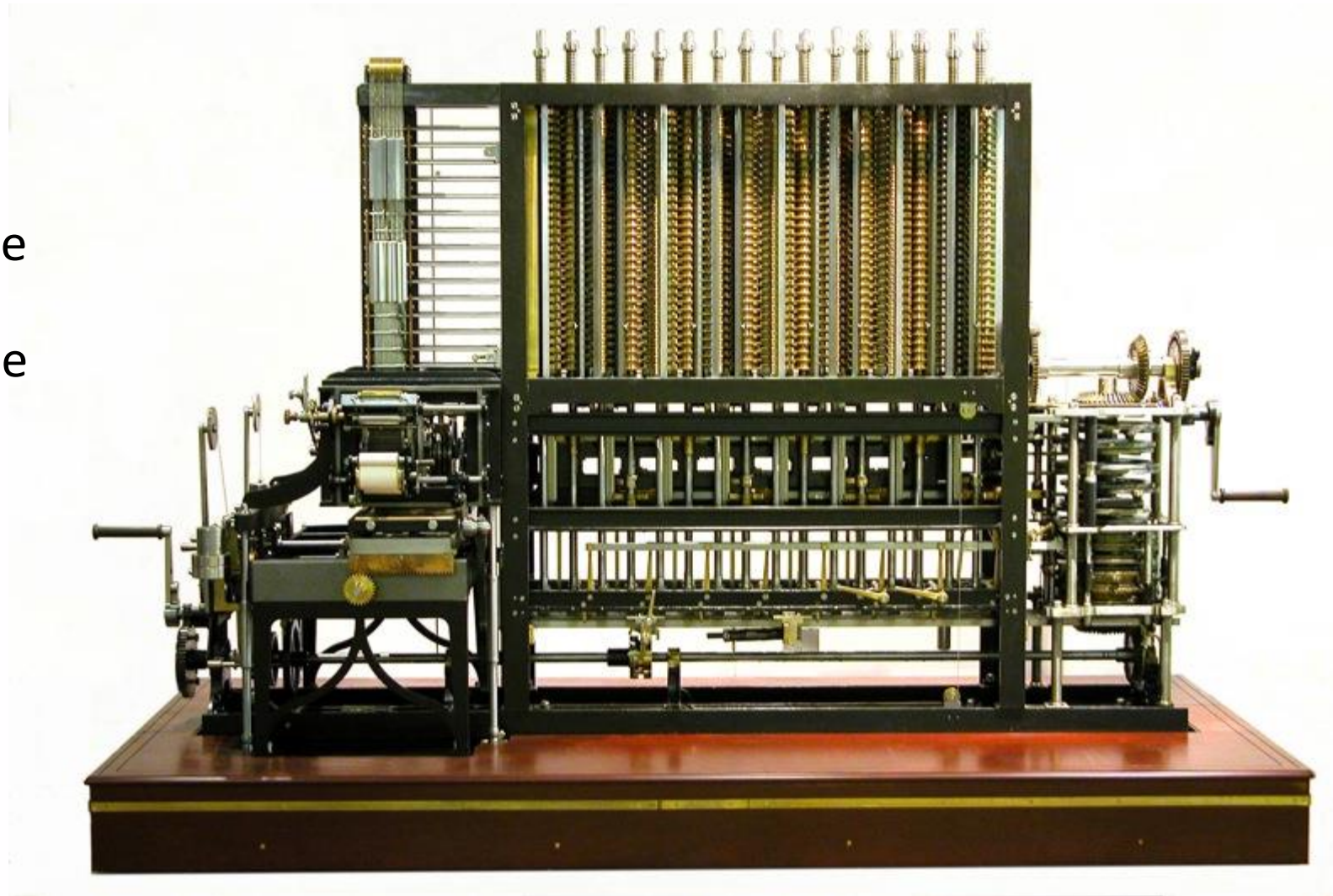
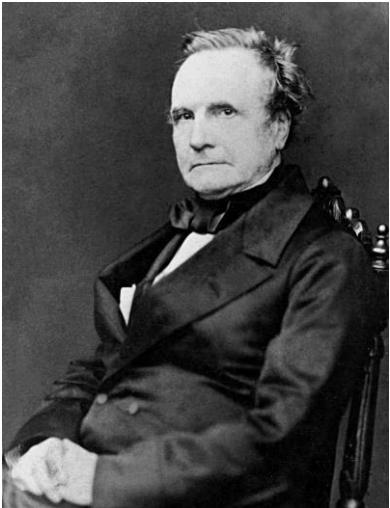


Mecanismo Jacquard no Musée des arts et métiers.  
(imagem: [David Monniaux](#))



## Breve Histórico

- Máquina de diferenças e máquina analítica de Charles Babbage (1822 e 1837).





## Breve Histórico

- Máquina de Hollerith (1890), de Hermann Hollerith
- Apuração do censo de 1890 em apenas 2 dias.
- O anterior, 1880 havia levado 7 anos.



Jennifer/flickr



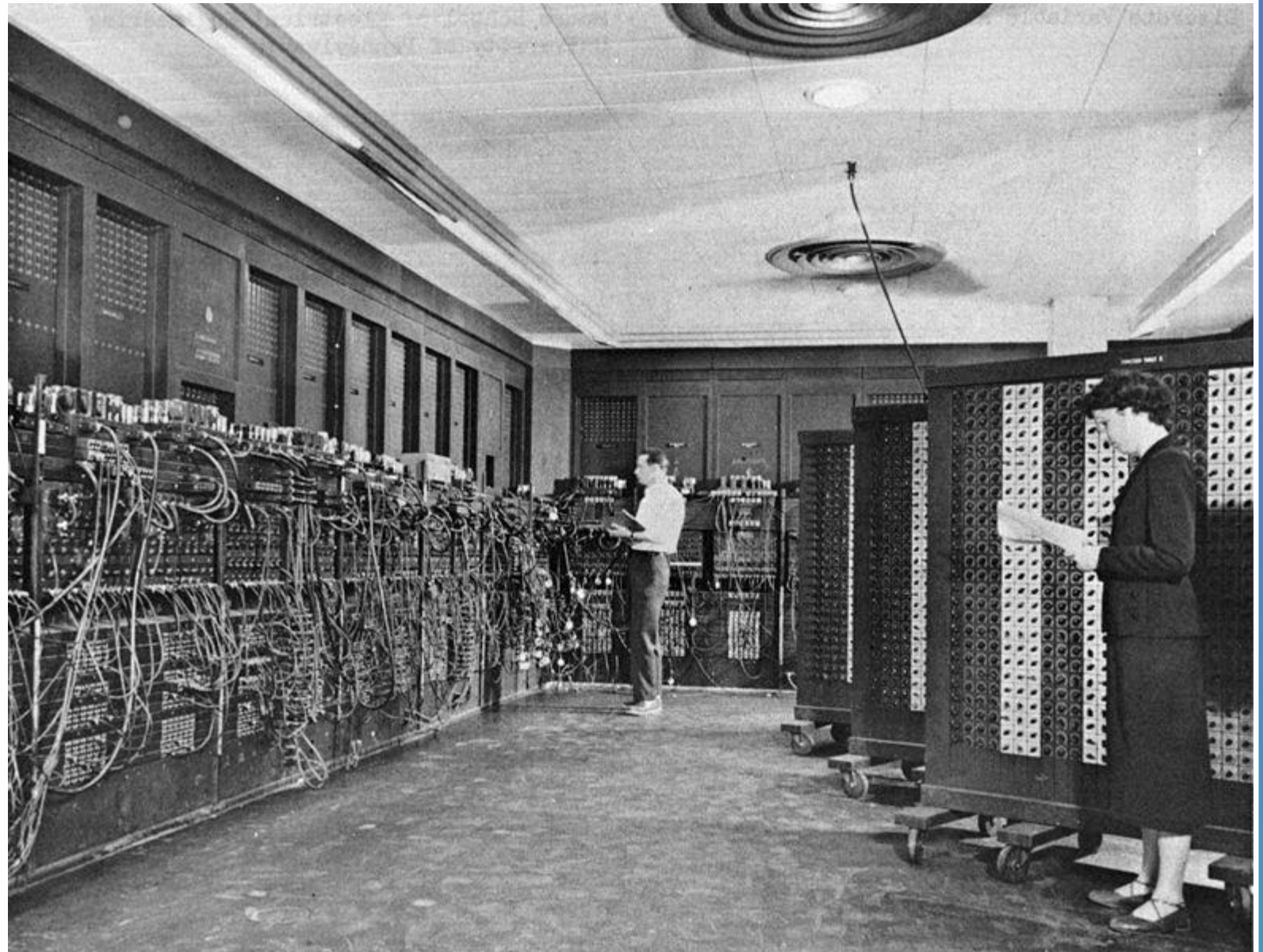
© picture-alliance/

picture-alliance/max

<https://www.dw.com/pt-br/1860-nasce-herman-hollerith-pai-do-processamento-de-dados/a-312287>

## Breve Histórico

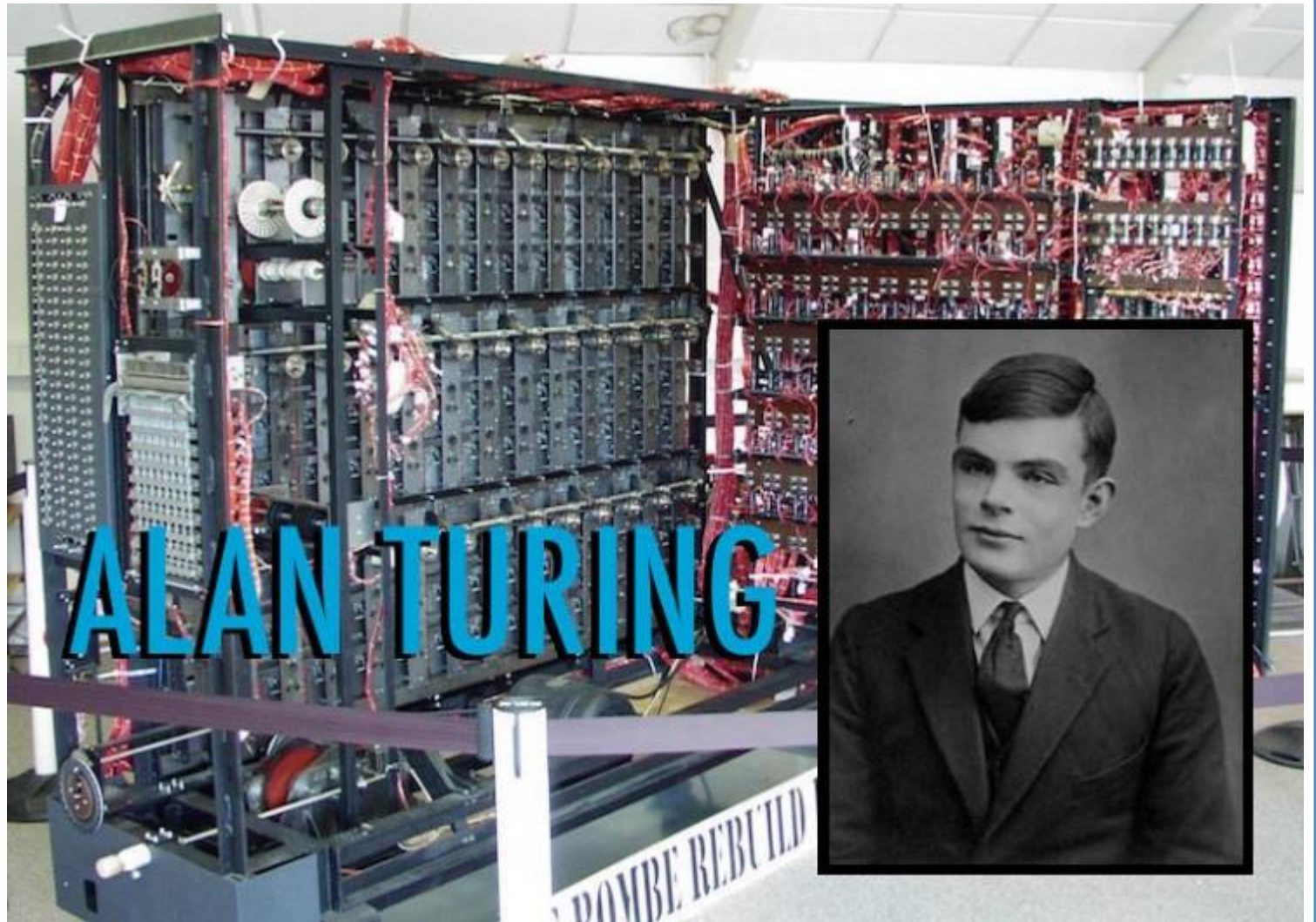
- John Von Neumann é conhecido principalmente por ter **formalizado o projeto lógico de um computador.**
- Computador ENIAC





## Breve Histórico

- Alan Turing.
- Com apenas 15 anos resolvia equações complexas. Aos 23 anos, formou-se em matemática pela com honras, pela Universidade de Cambridge.



Alan Turing trabalhou como funcionário do *Government Code and Cypher School* e entre 1940 e 1941, e **desenvolveu uma máquina capaz de decifrar o “Enigma”**, código utilizado pelos nazistas, durante a Segunda Guerra Mundial, dando assim aos aliados uma vantagem que permitiu derrotar mais depressa a Alemanha

[https://www.ebiografia.com/alan\\_turing/](https://www.ebiografia.com/alan_turing/)



# Breve História

- Todas as máquinas criadas até então eram dispositivos mecânicos e não utilizavam nenhum tipo de circuito elétrico.
- A utilização de circuitos elétricos só foi possível depois que um matemático chamado *George Boole* criou um sistema lógico no qual tudo poderia ser representado por meio de dois algarismos:
  - 0 ou 1.
  - 1 bit

# Lógica de Programação e Algoritmos

- Um **computador** nada mais é do que uma máquina com **circuitos eletrônicos capaz de executar um algoritmo**;
- Um **algoritmo** é um conjunto lógico de **operações predefinidas que resolva determinado problema**;
- De forma **intuitiva**, já **utilizamos algoritmos** em nosso dia a dia para realizar nossas tarefas;
- Em poucas palavras, um **algoritmo** é uma **receita para resolver um problema bem específico**.

# Lógica de Programação e Algoritmos

- Representação **gráfica**

- Conjunto de símbolos que representam todos os passos do algoritmo;
- Cada símbolo possui uma ação específica a ser executada;
- **Diagrama de Blocos.**

- Representação **textual**










- O português estruturado (**pseudocódigo**) é uma forma genérica (sem considerar linguagens de programação) de representar as ações do algoritmo.
- Iremos utilizar uma linguagem chamada **Pascal**



# Diagrama de Blocos

- Os principais símbolos utilizados em diagramas.

Os principais símbolos utilizados

Símbolo	Significado	Descrição
	Terminal	Representa a definição de início e fim do fluxo lógico a ser definido em um programa. É utilizado também na definição de sub-rotinas.
	Entrada manual	Representa a definição de entrada manual de dados, comumente realizada por meio de um teclado conectado ao computador.
	Processamento	Representa a definição da execução de ações de processamento normalmente relacionadas a operações matemáticas.
	Exibição	Representa a definição de execução da operação de saída de dados em um monitor de vídeo conectado ao computador.
	Decisão	Representa a definição de desvios condicionais nas operações de tomadas de decisões e laços condicionais para repetição de trechos de programa.
	Preparação	Representa a definição de execução de um laço incondicional que permite a modificação de instruções ou grupo de instruções limitadas no laço.
	Processo predefinido	Representa a definição de um grupo de operações relacionadas a uma sub-rotina de processamento.
	Conector	Representa pontos de conexão entre trechos de programas apontados a outras partes do diagrama de blocos.
	Linha	Representa os vínculos existentes entre os símbolos de um diagrama de blocos. Deve ter a ponta de uma seta indicando a direção que o fluxo de um programa deve seguir.

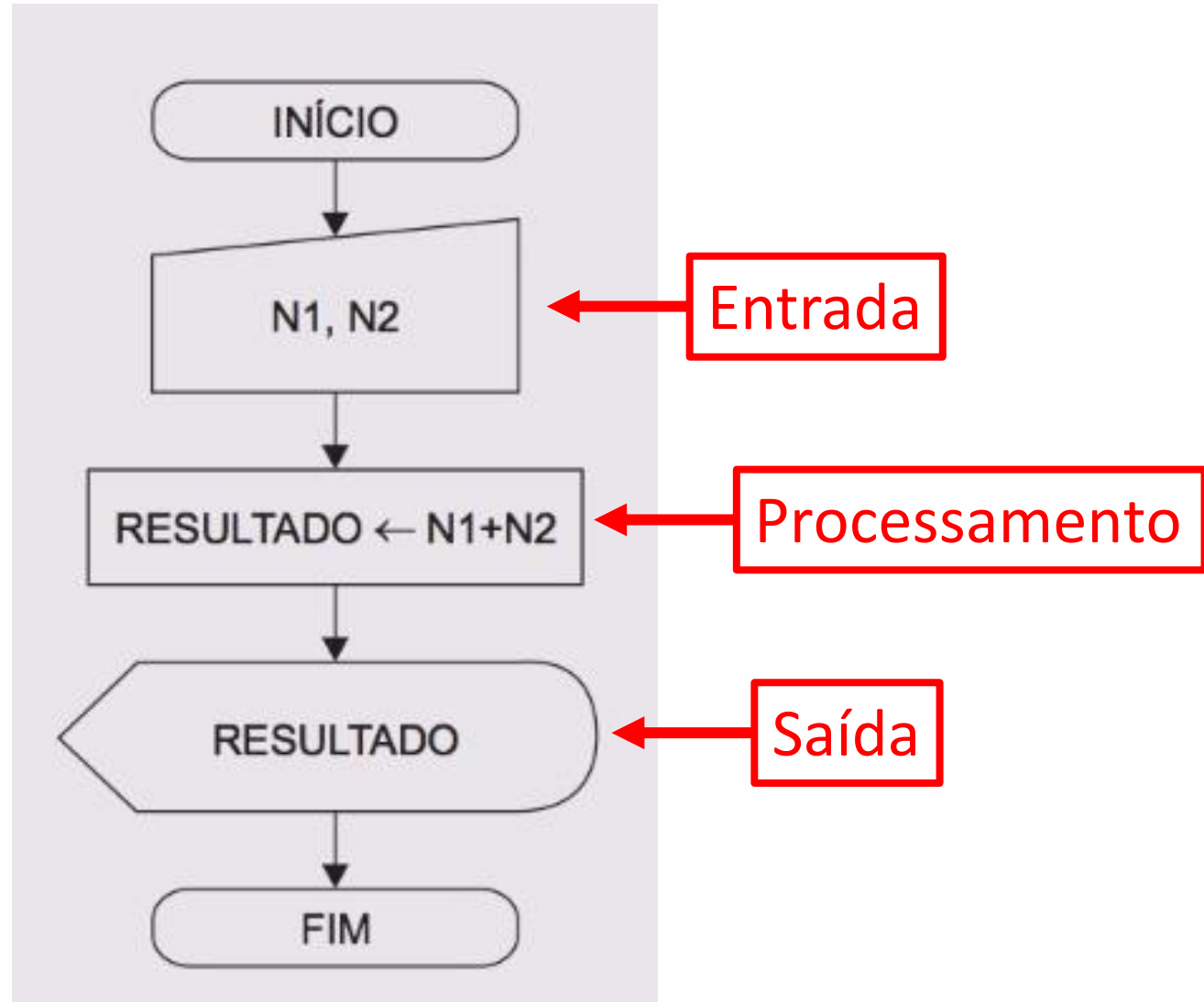
## Diagrama de Blocos

- Diagrama de bloco para calcular e imprimir a soma de dois números inteiros fornecidos pelo usuário



## Diagrama de Blocos

- Diagrama de bloco para calcular e imprimir a soma de dois números inteiros fornecidos pelo usuário





## Diagrama de Blocos

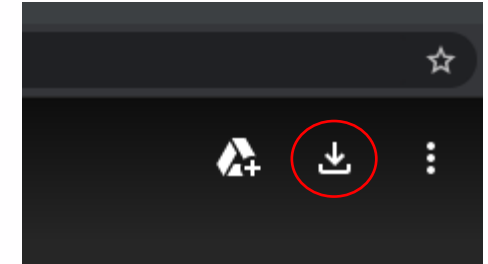
- Diagrama de bloco para calcular e imprimir a soma de dois números inteiros fornecidos pelo usuário



```
1 Program Soma;  
2 var  
3   n1, n2, resultado: integer;  
4 Begin  
5   readln (n1);  
6   readln (n2);  
7   resultado := n1 + n2;  
8   write(resultado);  
9 End.
```

# Ferramenta para a Linguagem Pascal

- Pascalzim:
  - <http://pascalzimbr.blogspot.com/p/blog-page.html>



## Pascalzim - Compilador Pascal

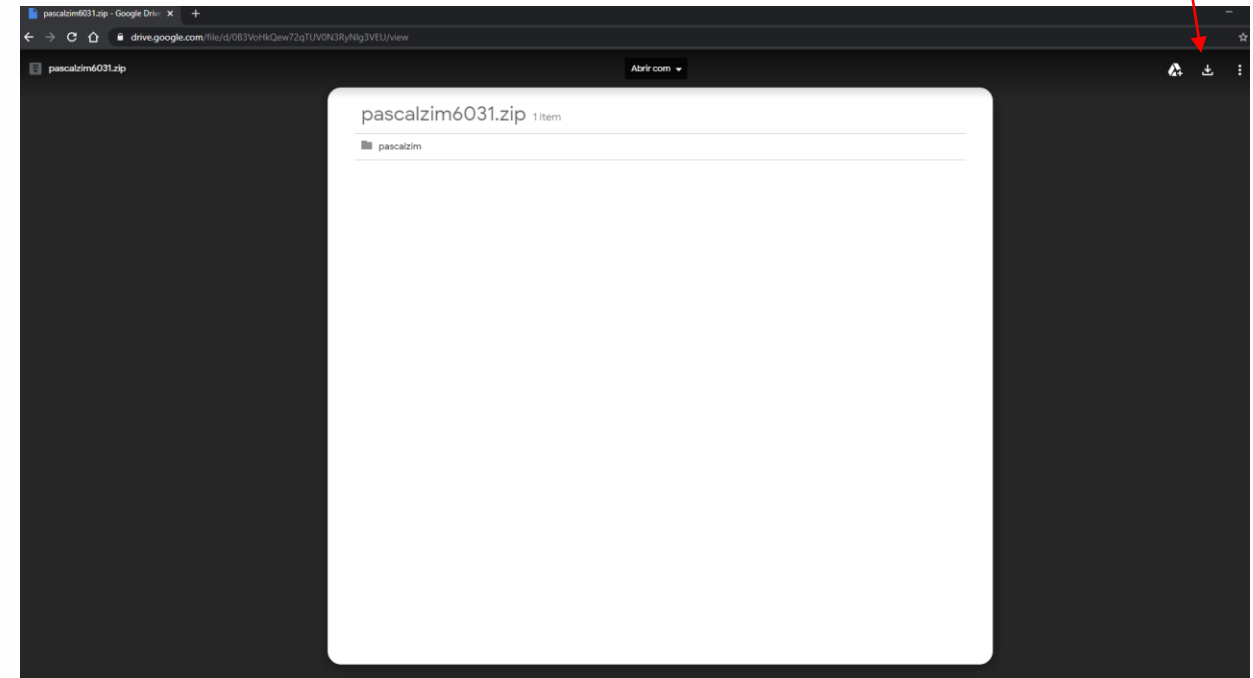
Download

**Versão atual do compilador:**  
Pascalzim 6.0.3 (19/02/2016)

**Requisitos Mínimos:**  
Sistema Operacional Windows







**Versões do compilador:**

- Pascalzim Versão 6.0.3.1
- Pascalzim Versão 6.0.3
- Pascalzim Versão 6.0.2
- Pascalzim Versão 6.0.1
- Pascalzim Versão 5.2.5
- Pascalzim Versão 5.2.4
- Pascalzim Versão 5.2.3
- Pascalzim Versão 5.2.2
- Pascalzim Versão 5.2.1
- Pascalzim Versão 5.2.0
- Pascalzim Versão 5.1.3
- Pascalzim Versão 5.1.2
- Pascalzim Versão 5.1.1
- Pascalzim Versão 5.1.0
- Pascalzim Versão 5.0.3



# Instalação

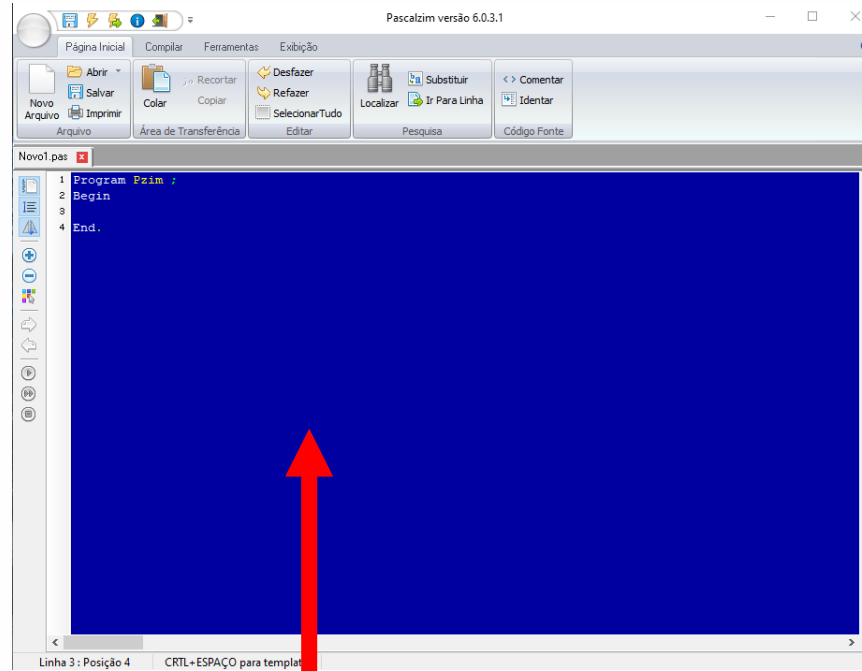
- Somente descompactar a pasta em algum local do seu computador, no caso, Windows.
- Para iniciar, é só executar o arquivo Pzim.exe.

Nome	Data de modificaç...	Tipo	Tamanho
 exemplos	19/06/2019 15:08	Pasta de arquivos	
 units	24/11/2016 09:14	Pasta de arquivos	
 leiame.txt	23/11/2016 13:43	Arquivo TXT	150 KB
 pzim.chm	15/02/2016 13:13	Arquivo de Ajuda ...	229 KB
 Pzim.exe	24/11/2016 09:00	Aplicativo	6.672 KB
 pzim.pdf	15/02/2016 13:13	Adobe Acrobat D...	1.052 KB

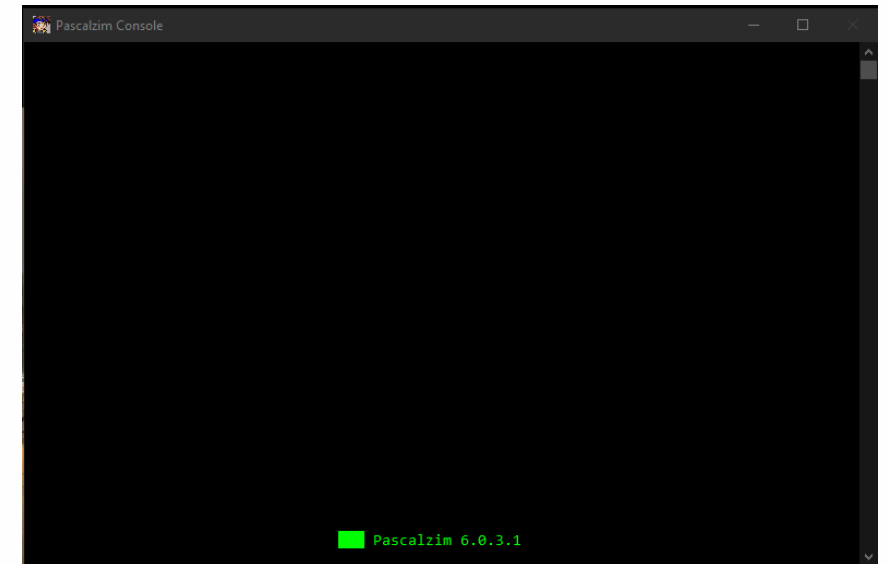


# O programa

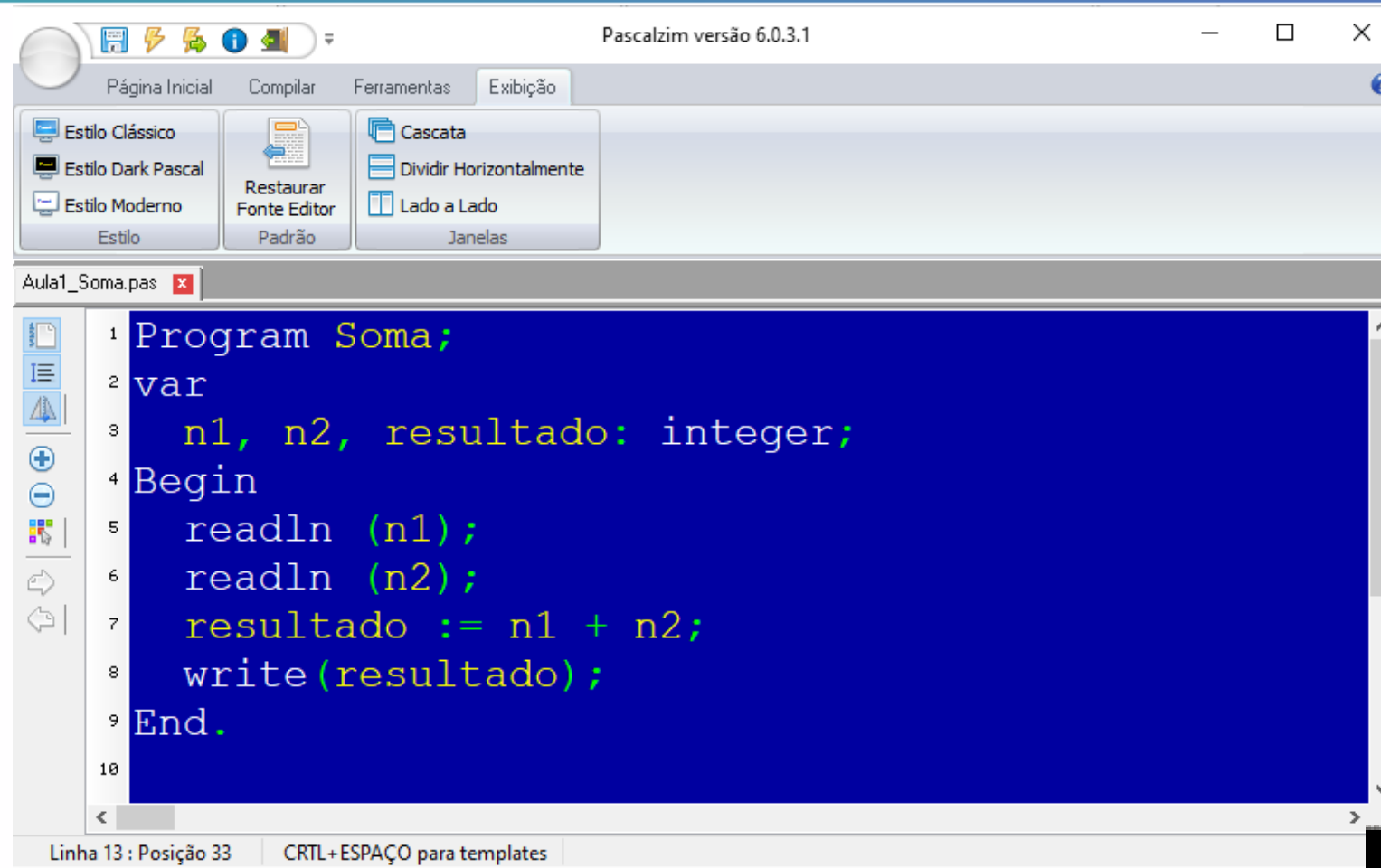
- Serão abertas 2 telas:  
1 para escrita do código (tela azul); e  
outra para as entradas e saídas do programa (Console – tela preta).



```
1 Program Soma;  
2 var  
3   n1, n2, resultado: integer;  
4 Begin  
5   readln (n1);  
6   readln (n2);  
7   resultado := n1 + n2;  
8   write(resultado);  
9 End.
```



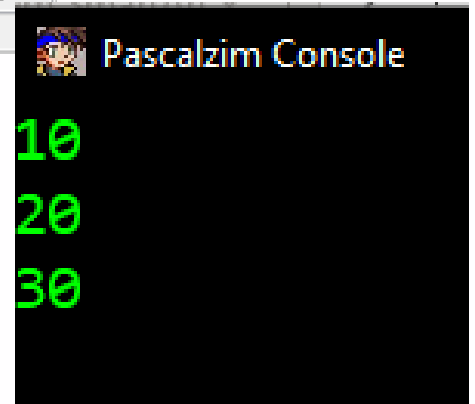
# O programa



The screenshot shows the Pascalzim 6.0.3.1 IDE. The title bar reads 'Pascalzim versão 6.0.3.1'. The menu bar includes 'Página Inicial', 'Compilar', 'Ferramentas', and 'Exibição'. The 'Exibição' menu is open, showing options for 'Estilo' (Estilo Clássico, Estilo Dark Pascal, Estilo Moderno) and 'Janelas' (Cascata, Dividir Horizontalmente, Lado a Lado). The main editor window displays the file 'Aula1\_Soma.pas' with the following Pascal code:

```
1 Program Soma;  
2 var  
3     n1, n2, resultado: integer;  
4 Begin  
5     readln (n1);  
6     readln (n2);  
7     resultado := n1 + n2;  
8     write(resultado);  
9 End.  
10
```

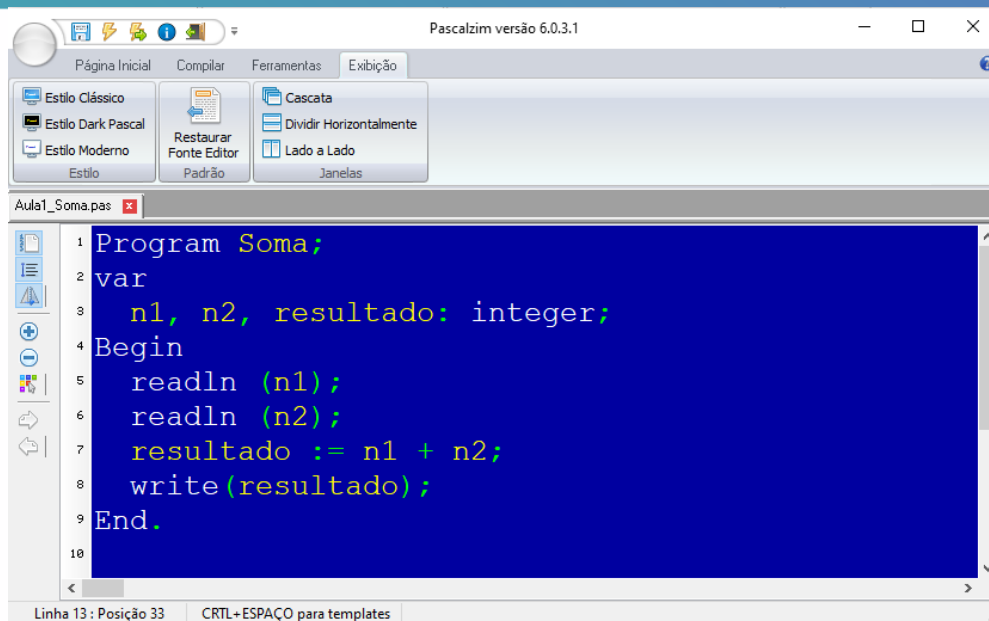
The status bar at the bottom indicates 'Linha 13 : Posição 33' and 'CTRL+ESPAÇO para templates'.



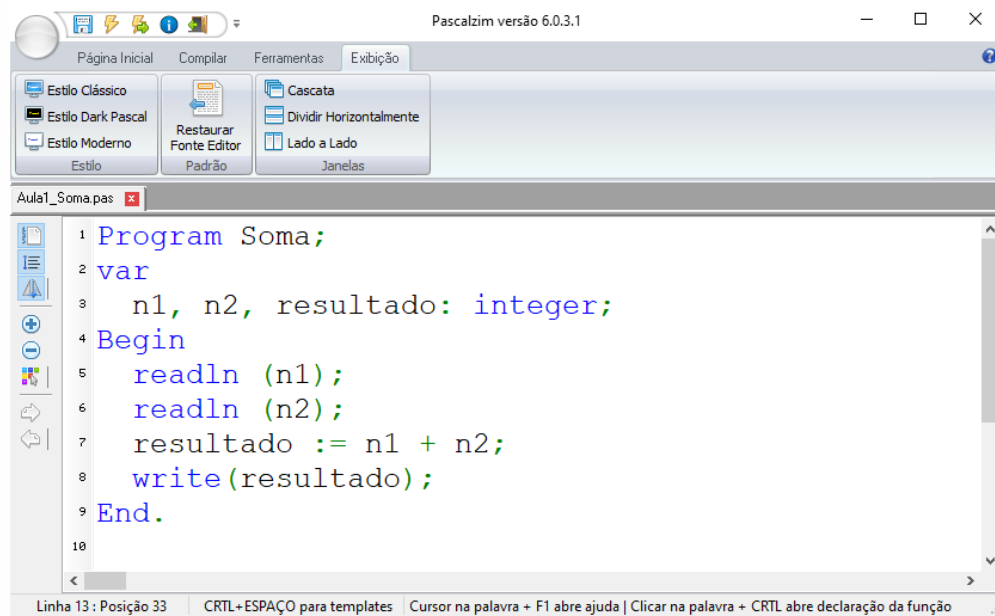
The screenshot shows the 'Pascalzim Console' window. It displays the output of the program, which consists of three lines of green text on a black background:

```
10  
20  
30
```

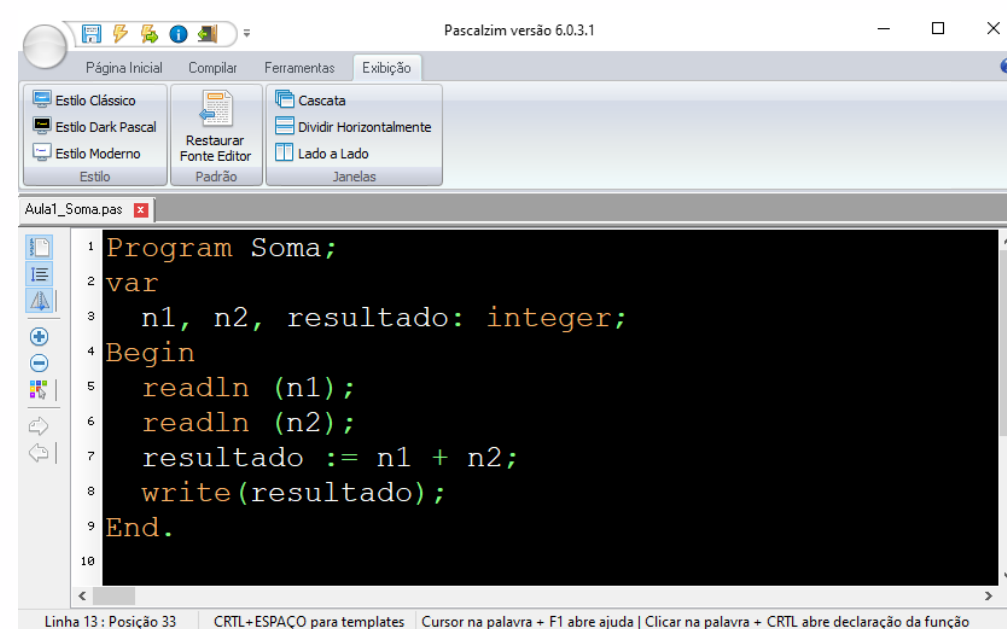
# O programa



```
1 Program Soma;
2 var
3   n1, n2, resultado: integer;
4 Begin
5   readln (n1);
6   readln (n2);
7   resultado := n1 + n2;
8   write(resultado);
9 End.
```



```
1 Program Soma;
2 var
3   n1, n2, resultado: integer;
4 Begin
5   readln (n1);
6   readln (n2);
7   resultado := n1 + n2;
8   write(resultado);
9 End.
```



```
1 Program Soma;
2 var
3   n1, n2, resultado: integer;
4 Begin
5   readln (n1);
6   readln (n2);
7   resultado := n1 + n2;
8   write(resultado);
9 End.
```

# Referências

- FORBELLONE, André L. **Lógica de Programação**. Prentice Hall Brasil, 3ª edição, 2005.
- VELOSO, Paulo; et al. **Estrutura de dados**. Rio de Janeiro: Campus, 4ª edição, 1996.
- LAGES & GUIMARAES. **Algoritmos e Estrutura de dados**. Ed. LTC, 1994.
- FARRER, H. **Algoritmos estruturados**. Rio de Janeiro: Guanabara Koogan, 3ª edição, 1989.
- LUIZ, Jaime. **Estrutura de dados e seus algoritmos**. Editora LTC.
- GUEDES, S. **Lógica de Programação Algorítmica**. Editora Pearson, 2014.
- MANZANO, José Augusto N. G. **Algoritmos lógica para desenvolvimento de programação de computadores**. Ed. 1, São Paulo, Erica 2016.
- MANZANO, José Augusto N. G. **Algoritmos técnicas de programação**. Ed 2, São Paulo, Erica, 2016.