

Curso de C

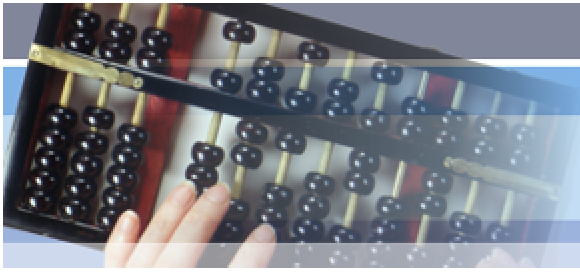


Introdução

3/10/2010
10:43 AM

Copyright © 2005--2009 by Arnaldo V. Moura e Daniel F. Ferber

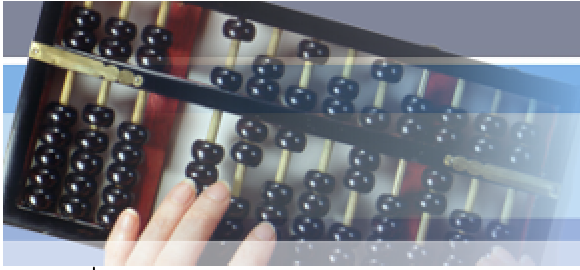
1



Introdução

Roteiro:

- Recordando Algoritmos
- Linguagem de Programação
- O computador
- Instruções de Máquina
- Níveis de Abstração
- Compilação



Algoritmos

Recordando:

Algoritmo: conjunto finito de instruções

- Usualmente, começa com a primeira instrução
- Execução seqüencial, uma instrução de cada vez, com possibilidade de saltos para outras instruções
- Instruções individuais suficientemente elementares, ou primitivas
- Sempre deve alcançar uma instrução PARE, para terminar a execução do algoritmo.

Utiliza dados (**entrada**) e gera um resultado (**saída**)

Algoritmos

Exemplo: Calcular o máximo divisor comum

Algoritmo em Português

Entrada

1. Leia dois números.
2. Divida o primeiro pelo segundo e guarde o resto.
3. Se o resto for diferente de 0 (zero),
então salta para passo 5
4. Escreva o segundo número e PARE.
5. Substitua o primeiro número pelo segundo.
6. Substitua o segundo número pelo resto da divisão.
7. Retorne ao passo 2.

Condição

Saída

Salto

Início

Fim



Algoritmos

- **Cada instrução precisa ser:**
 - Não ambígua
 - Uma única operação bem definida
 - Tecnicamente viável
- **Representações possíveis:**
 - Diagramas
 - Modelos matemáticos
 - Linguagens de programação

Algoritmos

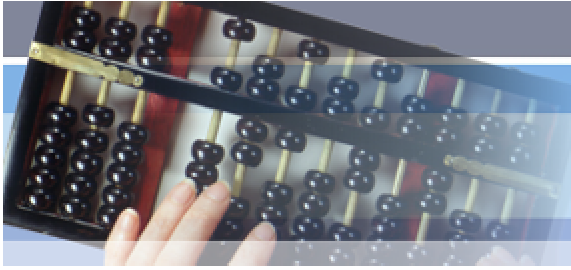
Exemplo: Calcular o máximo divisor comum

Algoritmo em Português

1. Leia dois números.
2. Divida o primeiro pelo segundo e guarde o resto.
3. Se o resto for diferente de 0 (zero), então salta para passo 5
4. Escreva o segundo número e PARE.
5. Substitua o primeiro número pelo segundo.
6. Substitua o segundo número pelo resto da divisão.
7. Retorne ao passo 2.

Potencialmente ambíguo

Não é uma única instrução bem definida



Linguagem de Programação

Opções de representar algoritmos:

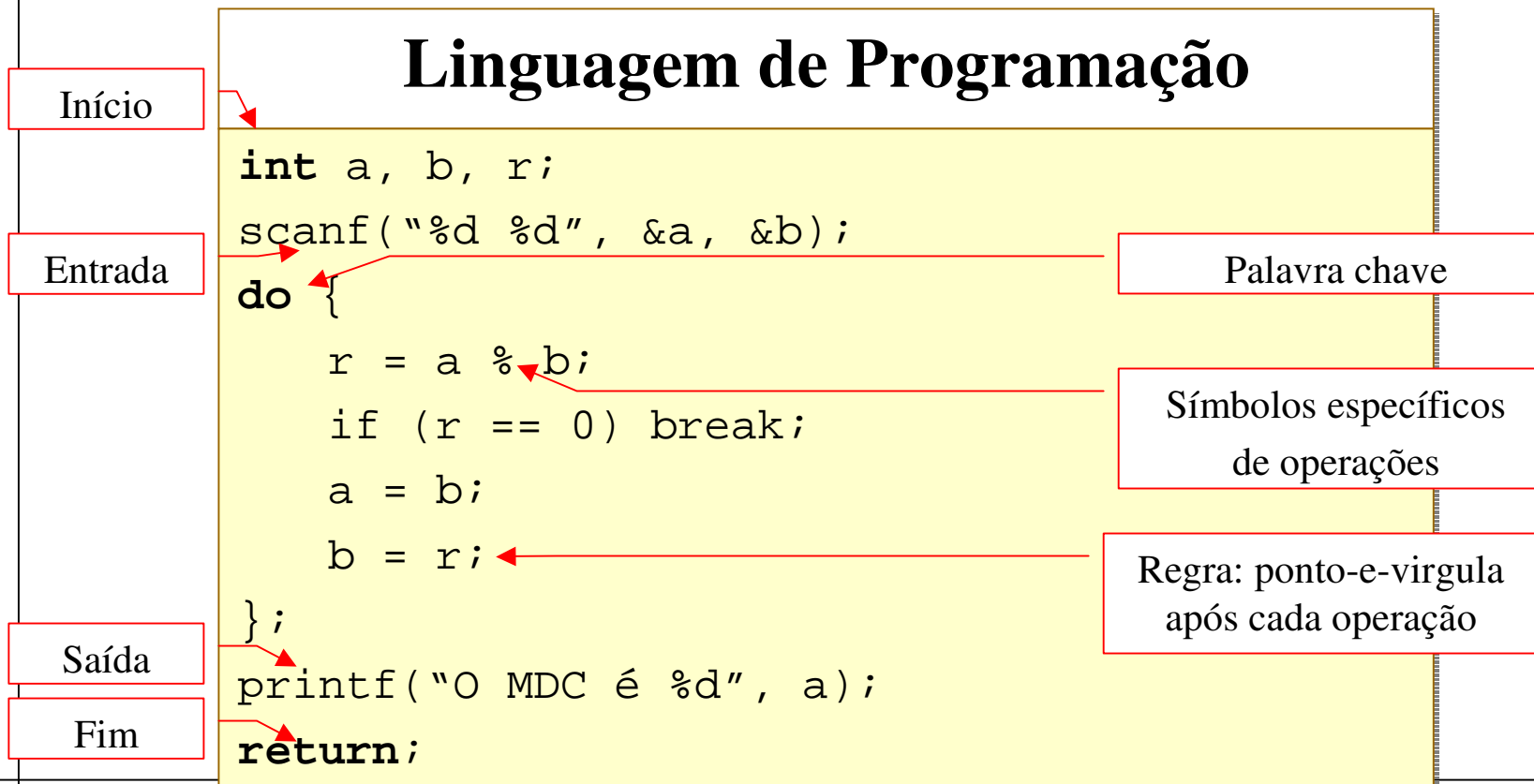
- Diagramas
- Uma linguagem específica para escrever algoritmos
- Linguagens de programação

Linguagem de programação

- **Conceito:**
 - A linguagem de programação é um veículo para se escrever algoritmos.
- **Características:**
 - Vocabulário restrito
 - Regras de sintaxe
 - Verificação automática da sintaxe

Exemplo

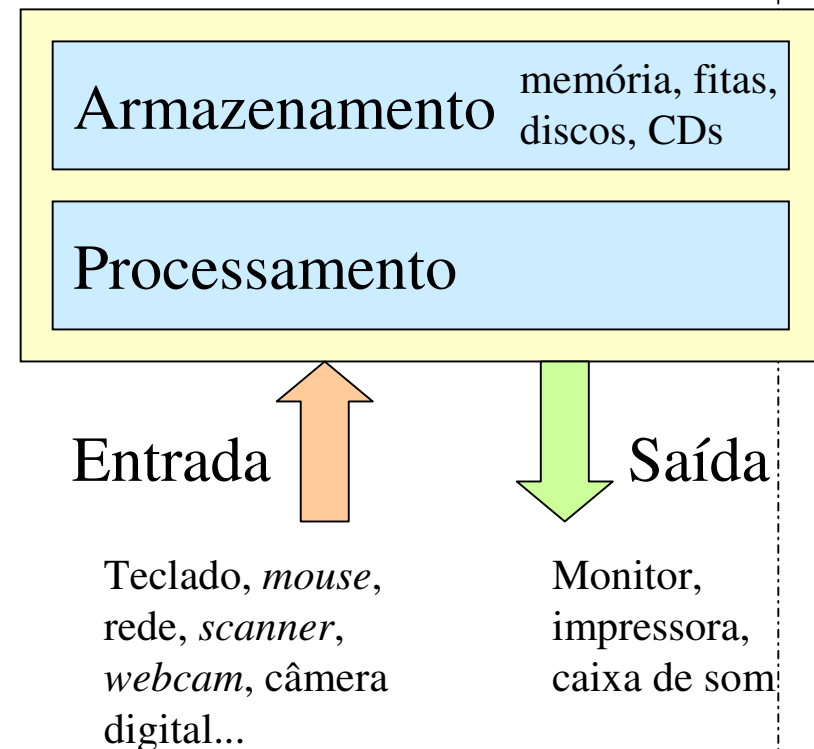
Exemplo: Calcular o máximo divisor comum



O Computador

Conceito:

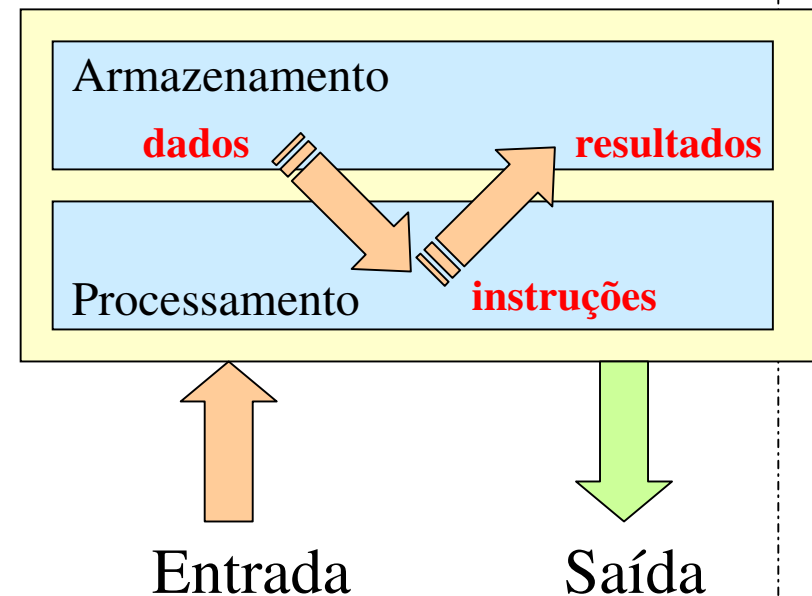
- Realiza **processamento** sobre dados **armazenados** no computador.
- Executa **operações matemáticas** e **lógicas** sobre dados.
- Recebe dados do meio externo (**entrada**)
- Apresenta os resultados para o meio externo (**saída**)



O Computador

Instruções de Máquina:

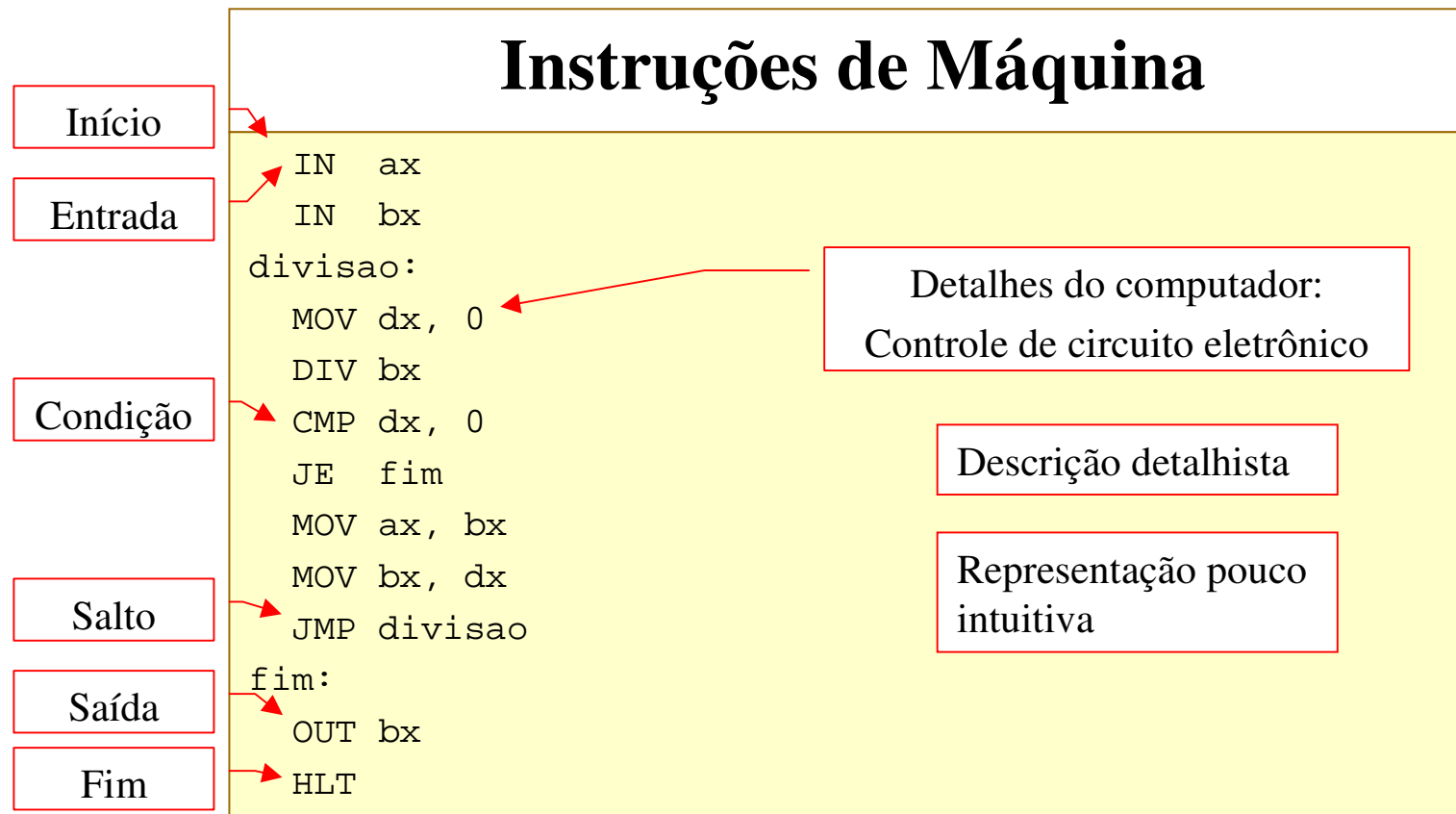
- Representam operações primitivas sobre dados.
- Em formato digital próprio (código de máquina)



O computador é uma máquina **rápida** e **eficiente**
para **simular algoritmos!**

O Computador

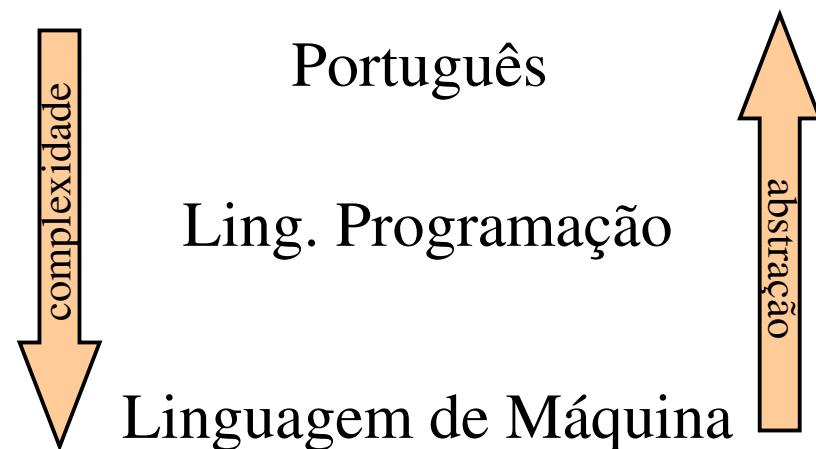
Exemplo: Calcular o máximo divisor comum



Níveis de Abstração

Dois casos extremos:

- Português:
 - Fácil, intuitivo
 - Computador não entende
 - Ambíguo, mal definido
- Linguagem de Máquina:
 - Complexo e trabalhoso
 - Única forma aceita pelo computador
 - Preciso, bem definido
 - Envolve detalhes específicos do computador, irrelevantes para o algoritmo



Níveis de Abstração

Objetivos:

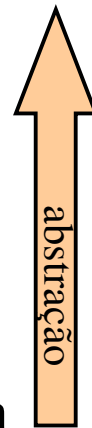
- Descrição precisa do algoritmo
- Independente do computador
- Nível de complexidade intermediário!



Português

Ling. Programação

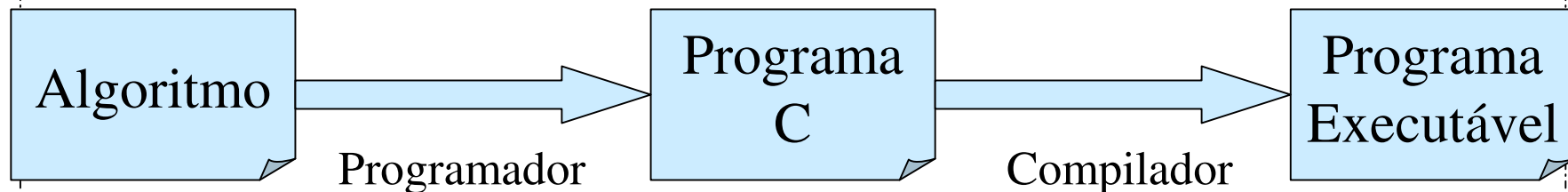
Linguagem de Máquina



Linguagem de programação

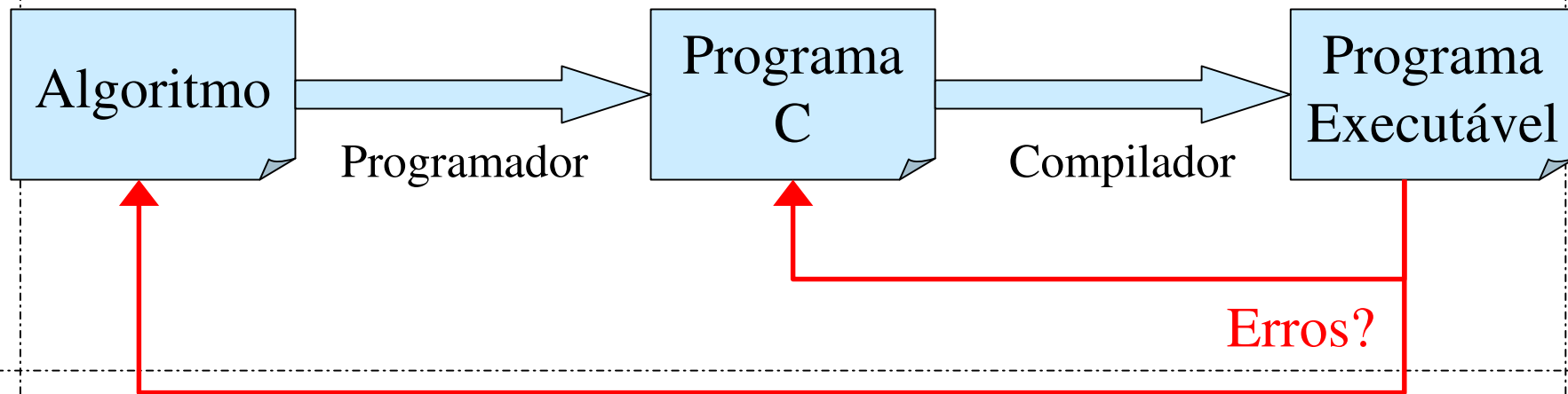
Construção do Programa

- Passo 1: Elaborar um algoritmo
- Passo 2: Reescrever o algoritmo em C
- Passo 3: Acionar o compilador



Construção do Programa

- Passo 4: Testar o programa
- Erros?** Verificar programa em C
- Continuam erros?** Verificar o algoritmo!



Curso de C

Primeiro Programa

3/10/2010
10:43 AM

Copyright © 2005--2009 by Arnaldo V. Moura e Daniel F. Ferber

17

Primeiro Programa

Roteiro:

- O primeiro programa em C
- Estrutura do código fonte
 - Comentários
 - Diretivas de compilador
 - Procedimento principal
- Estilo do código fonte

Exemplo

O Programa “Bom Dia”:

```
/* PrimeirosPassos.c: Nosso primeiro programa em C */  
  
#include <stdio.h> // mais um comentario  
#include <stdlib.h>  
  
int main(int argc, char* argv[]) {  
    printf("O primeiro programa lhe deseja um bom dia!");  
    return 0;  
}
```

Estrutura do Código Fonte

Comentários

Diretivas de
compilador

Procedimento
principal

Instrução

Instrução

Pontuação

```
/* PrimeirosPassos.c */  
  
#include <stdio.h>  
#include <stdlib.h>  
  
int main(int argc, char* argv[]) {  
    printf("bom dia!");  
    return 0;  
}
```

Estrutura do Código Fonte

Comentários:

- Texto ignorado pelo compilador
- Documentação útil para descrever trechos do algoritmo
- Possível em qualquer posição do código fonte
- Duas formas para comentários:
 - Uma linha: `// Comentário ...`
 - Várias linhas: `/* Comentário...`

`mais comentários ... */`

```
// PrimeirosPassos.c: Nosso primeiro programa em C
```

```
/* PrimeirosPassos.c: Nosso primeiro programa em C */
```

```
/* PrimeirosPassos.c:  
Nosso primeiro programa em C */
```

Estrutura do Código Fonte

Diretivas de Compilador:

- Informam outros arquivos que devem ser consultados antes de compilar
- Definem parâmetros utilizados pelo compilador
- Colocadas no início do código fonte

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
```

Estrutura do Código Fonte

Procedimento principal:

- Seqüência de instruções
- Pontuação: ponto-e-vírgula termina instruções
- Chaves agrupam instruções relacionadas

```
int main(int argc, char* argv[]) {  
    printf("bom dia!");  
    return 0;  
}
```

Estrutura do Código Fonte

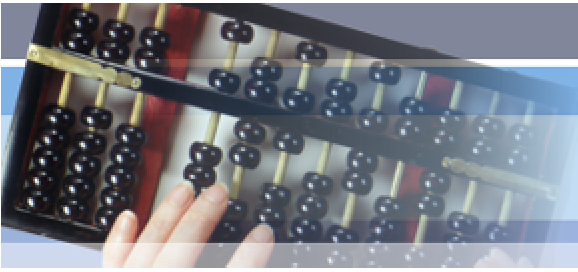
Procedimento principal:

- Siga sempre o seguinte esqueleto:

```
int main(int argc, char* argv[]) {  
    ...  
    Algoritmo  
    ...  
    return 0;  
}
```

Obrigatório

Obrigatório



Estrutura do Código Fonte

Práticas interessantes:

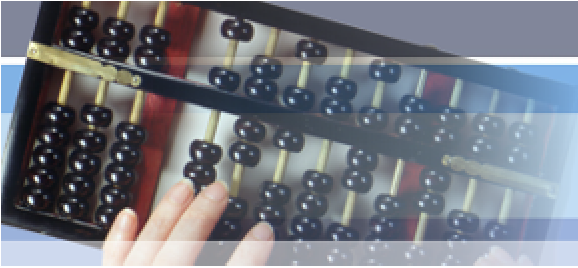
- Linhas em branco são ignoradas

```
int main(int argc, char* argv[]) {  
    printf("bom dia!");  
    return 0;  
}
```

(recomendado)

```
int main(int argc, char* argv[]) {  
  
    printf("bom dia!");  
  
    return 0;  
}
```

(permitido)



Estrutura do Código Fonte

Práticas interessantes:

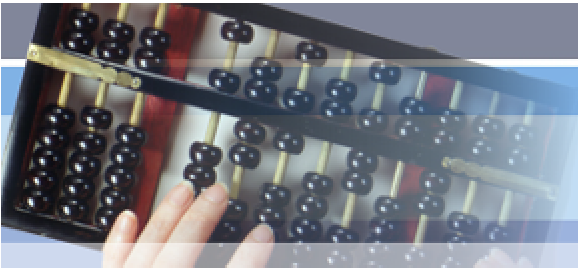
- Espaços e tabulações são ignoradas

```
int main(int argc, char* argv[]) {  
    printf("bom dia!");  
    return 0;  
}
```

(recomendado)

(permitido,
mais confuso)

```
int main(int argc, char* argv[]) {  
printf("bom dia!");    return 0;  
}
```



Estrutura do Código Fonte

Práticas interessantes:

- De preferência, uma instrução por linha

```
int main(int argc, char* argv[]) {  
    printf("bom dia!");  
    return 0;  
}
```

(recomendado)

(permitido,
mais confuso)

```
int main(int argc, char* argv[]) {  
    printf("bom dia!"); return 0;  
}
```

A background image showing a person's hands using a traditional abacus. The abacus is black with wooden rods and black beads. The person's fingers are visible, moving the beads. The image is semi-transparent and serves as a background for the text.

Curso C

*Primeiro Programa no
Microsoft Visual Studio .net*

Primeiro Programa

Roteiro:

- Criar um novo projeto
- Escrever o programa
- Compilar
- Executar

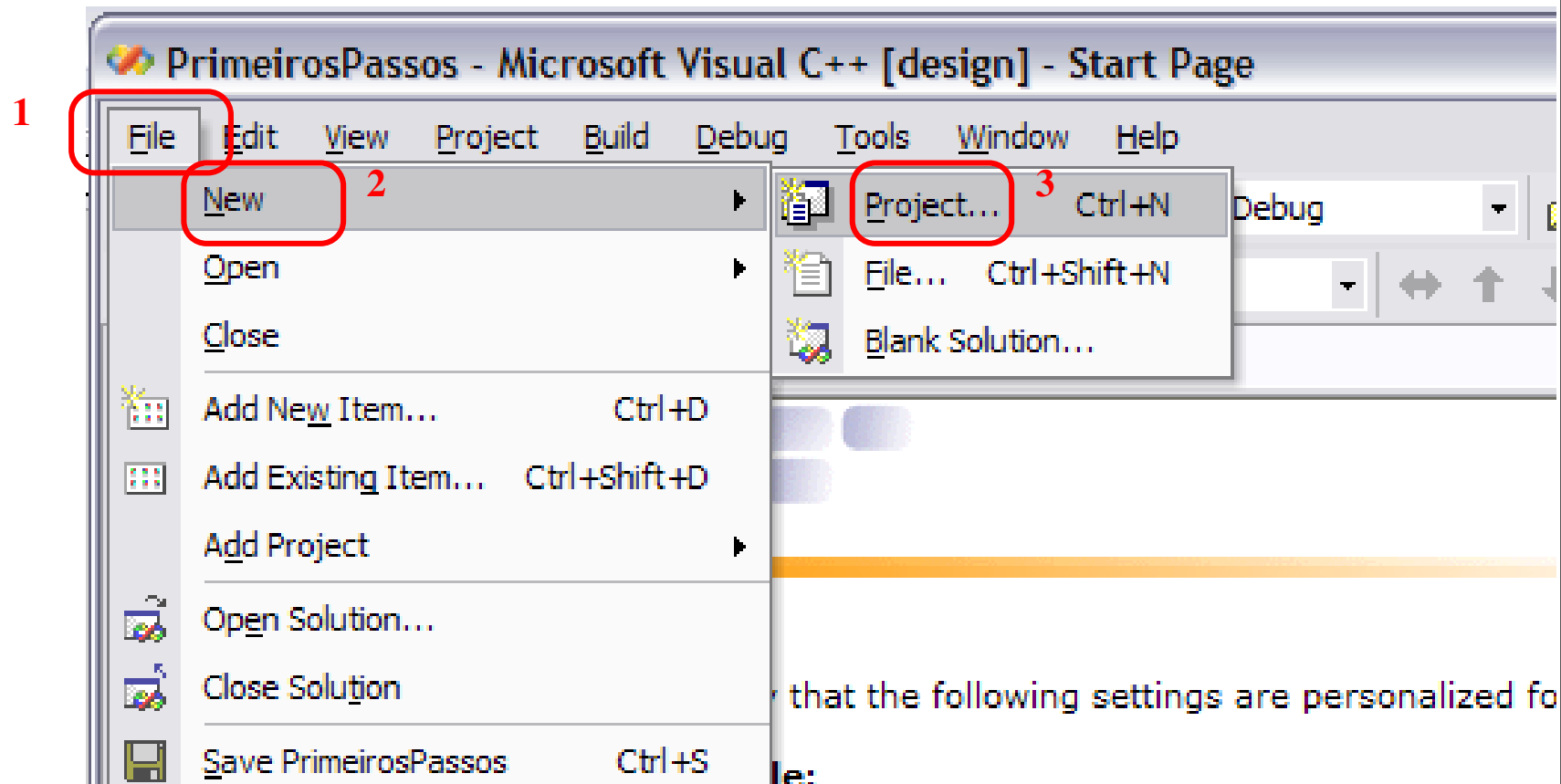
Primeiro Programa

Projeto Visual Studio:

- Criar um projeto para cada programa!
- Um projeto contém:
 - Arquivos em Linguagem de Programação
 - Configuração
 - Programa compilado
- Projeto:
 - tipo e recursos disponíveis:
 - Interface Gráfica
 - Ferramentas adicionais de programação

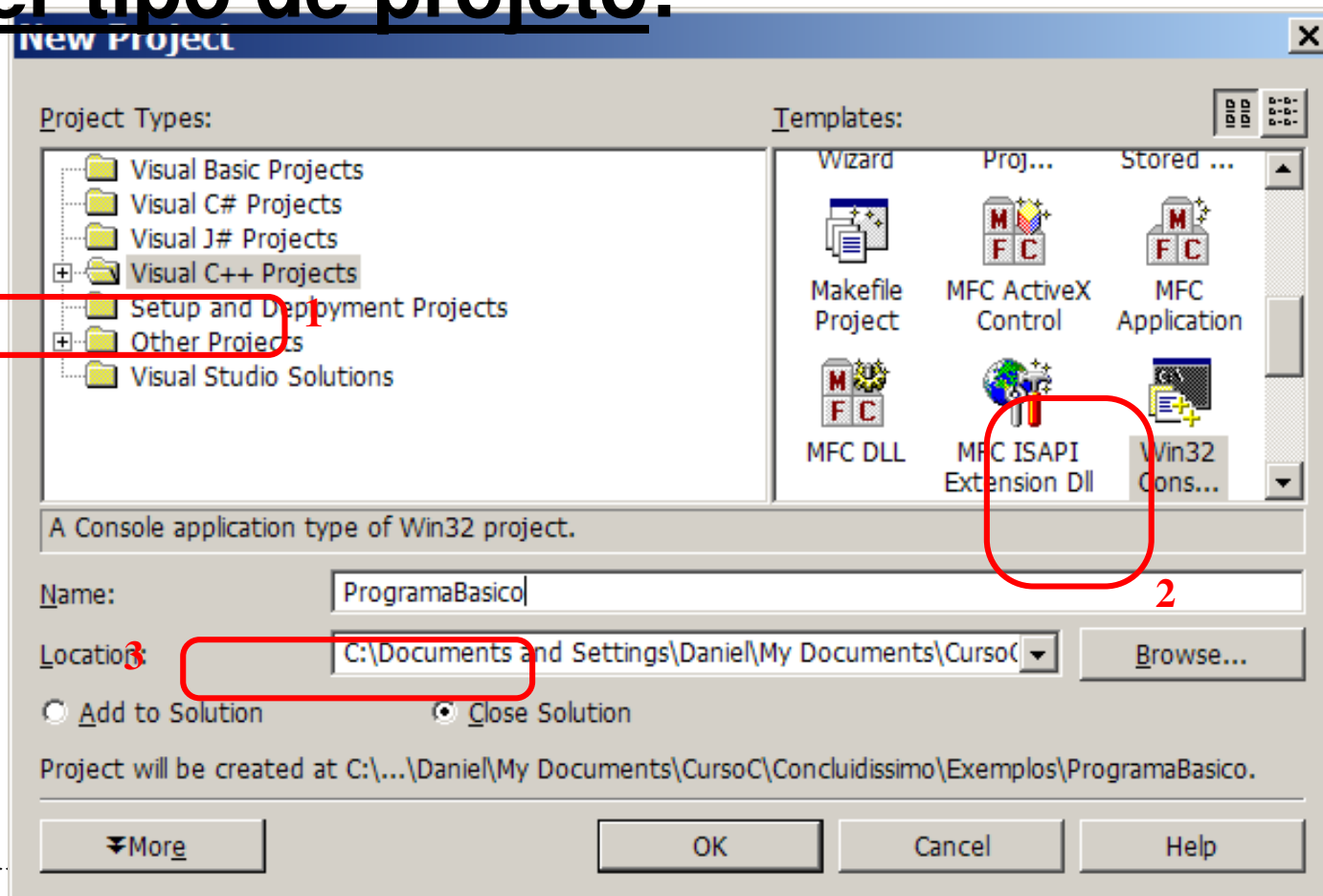
Primeiro Programa

Criar projeto:



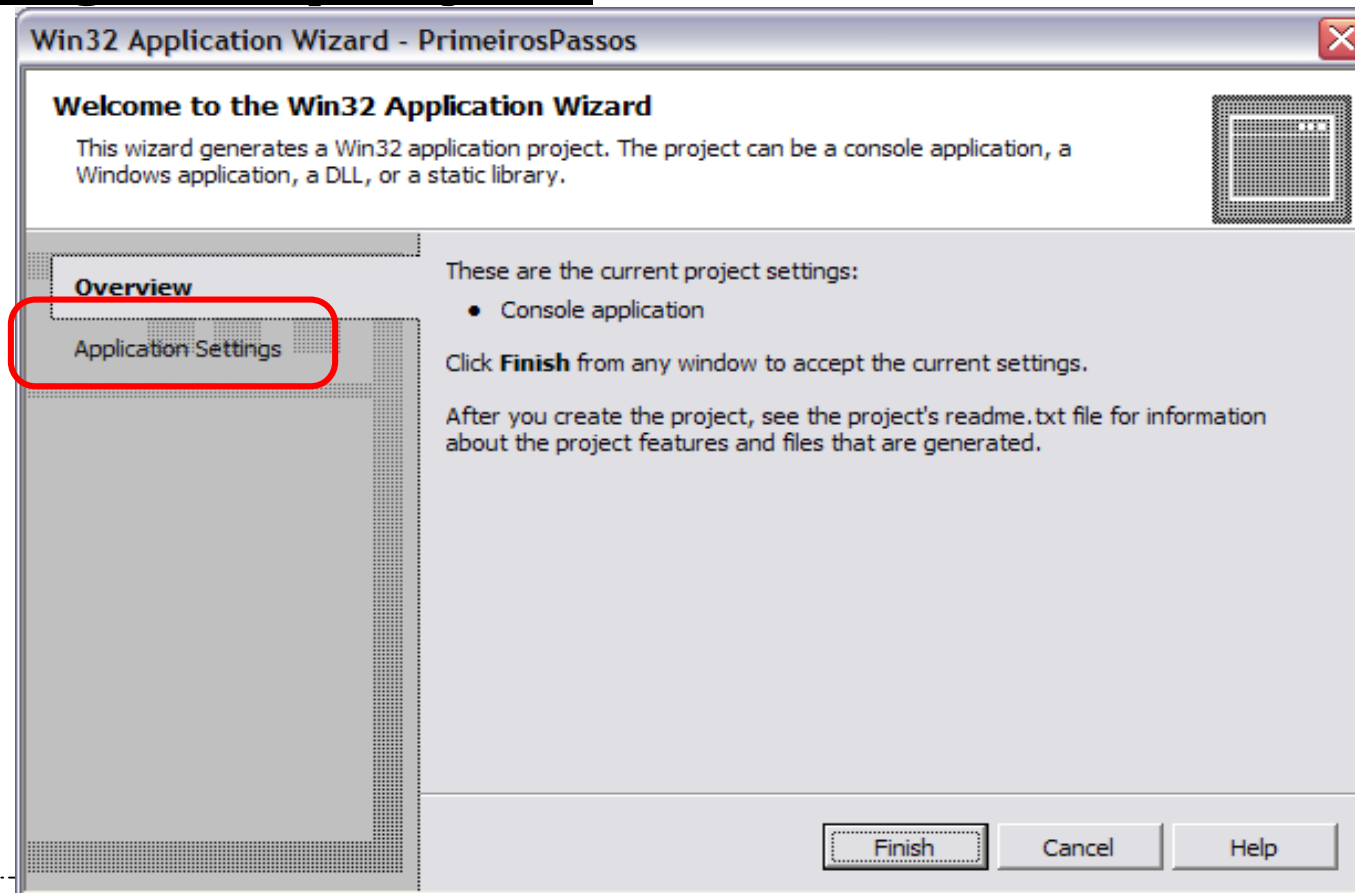
Primeiro Programa

Escolher tipo de projeto:



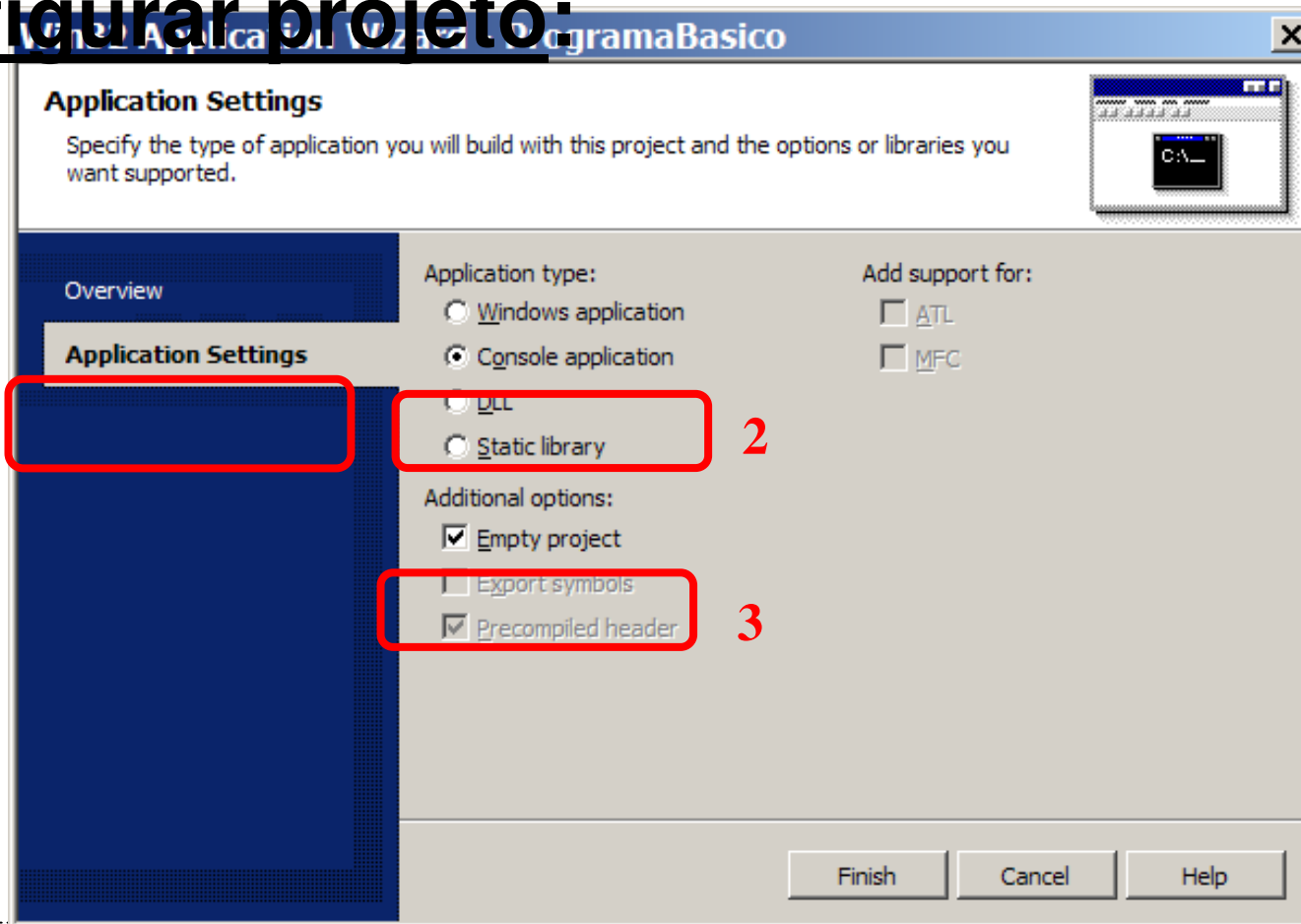
Primeiro Programa

Configurar projeto:



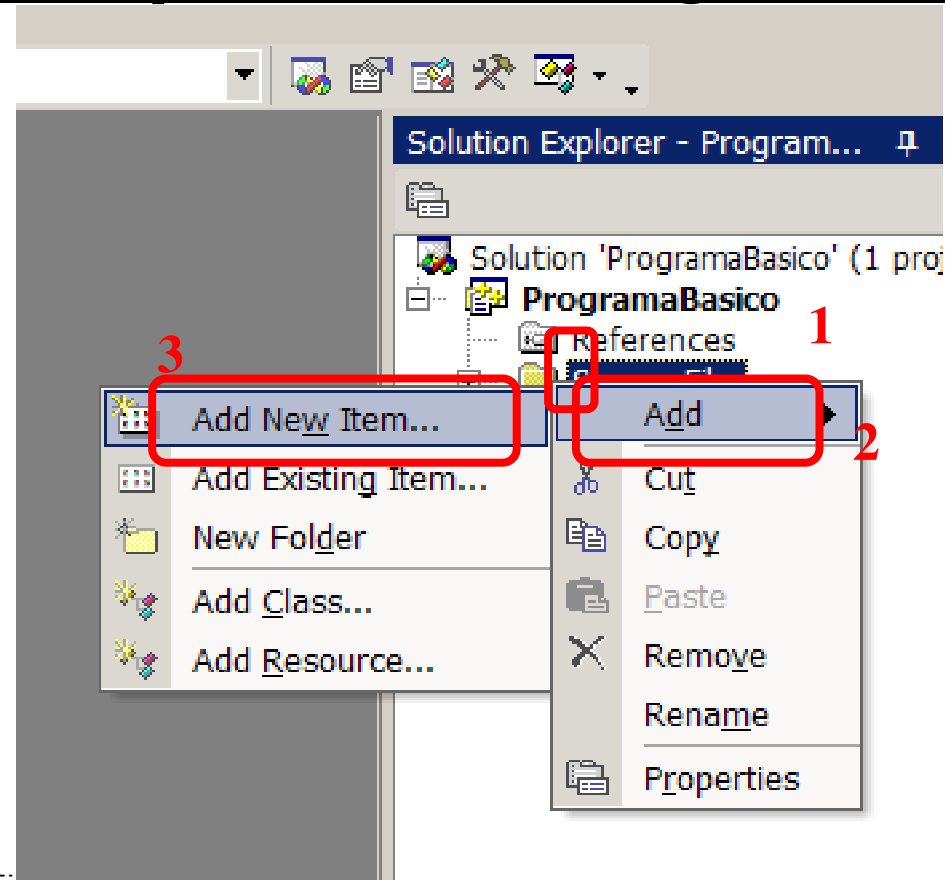
Primeiro Programa

Configurar projeto:



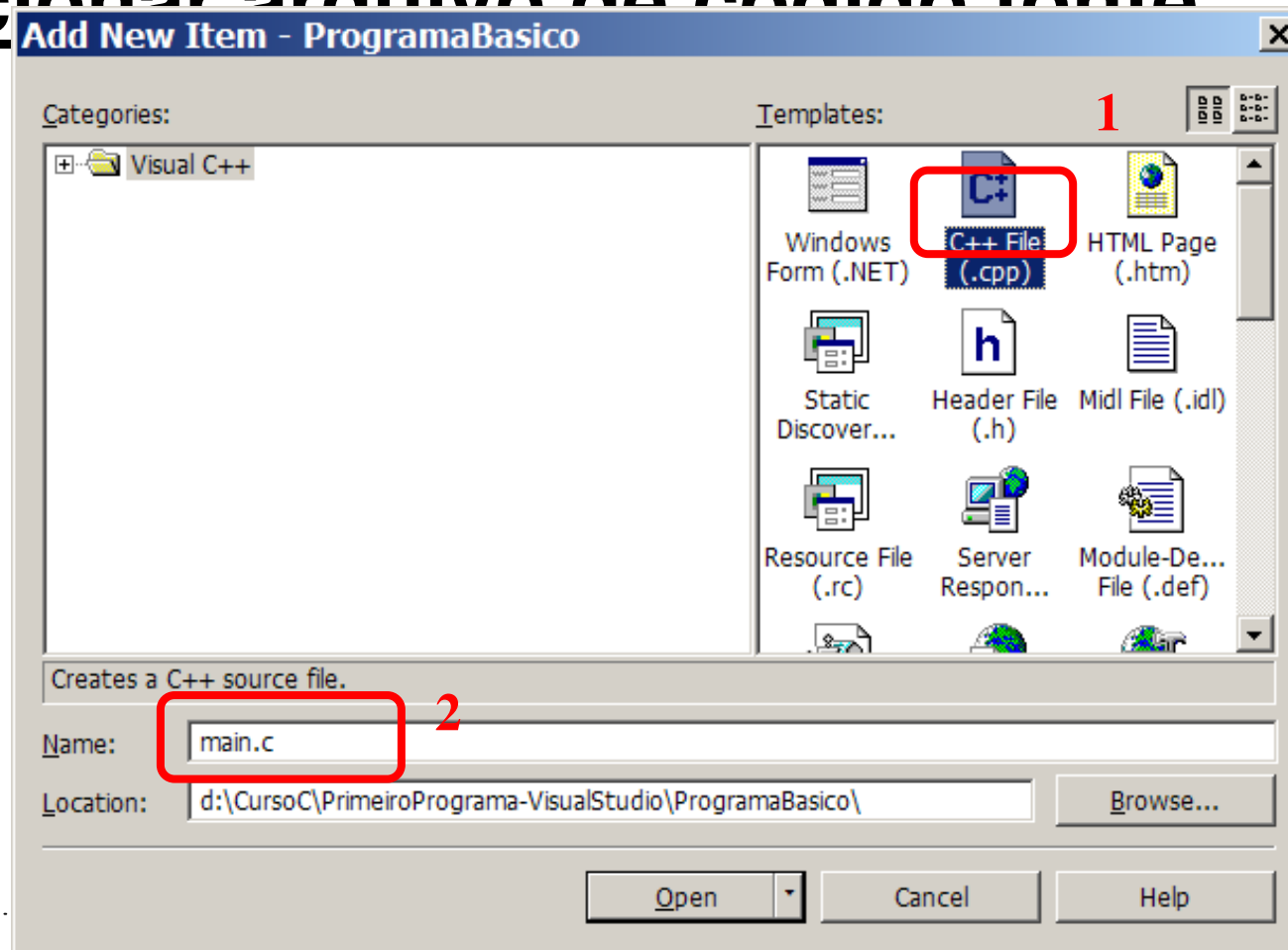
Primeiro Programa

Adicionar arquivo de código fonte:



Primeiro Programa

Adicionar arquivo de código fonte:

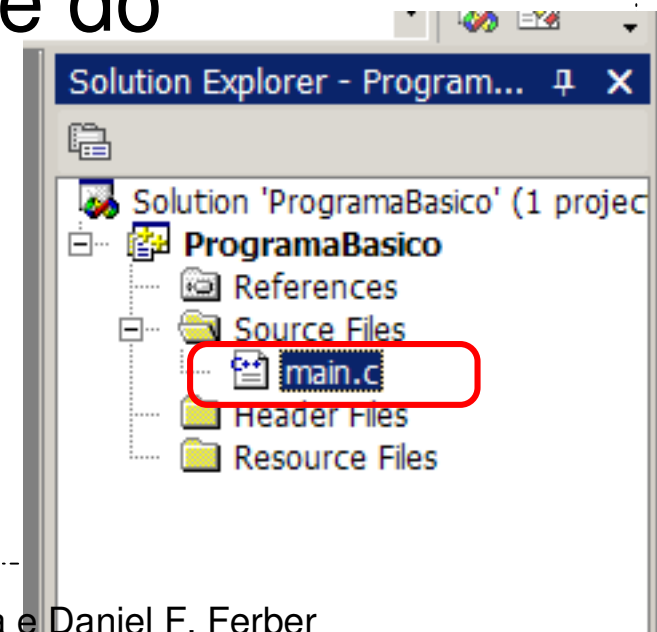


Primeiro Programa

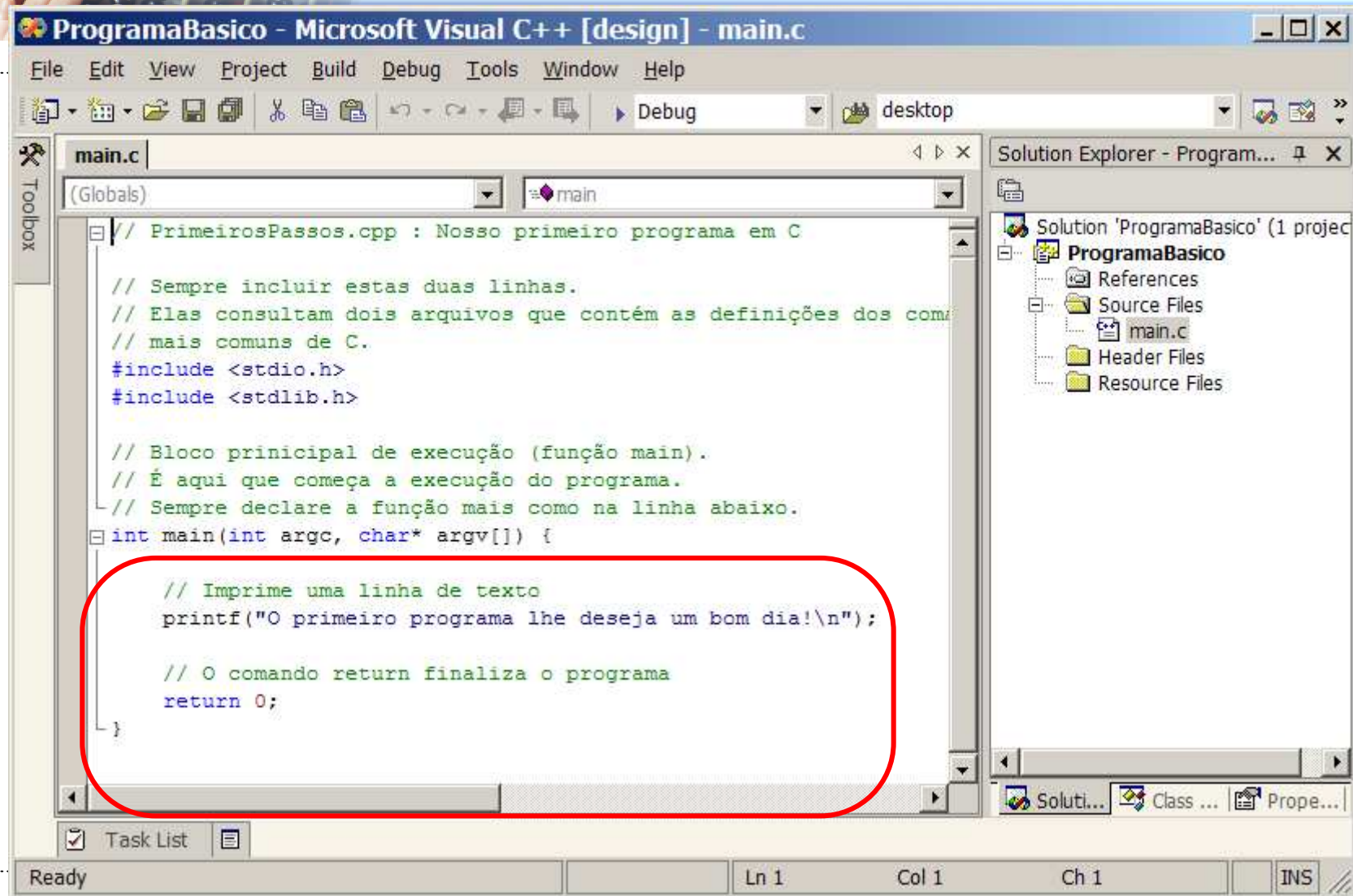
Editar código fonte:

Abrir o arquivo que armazenará o código fonte:

1. Duplo-clicar sobre o nome do arquivo: "main.c"
2. O conteúdo do arquivo é mostrado na tela.



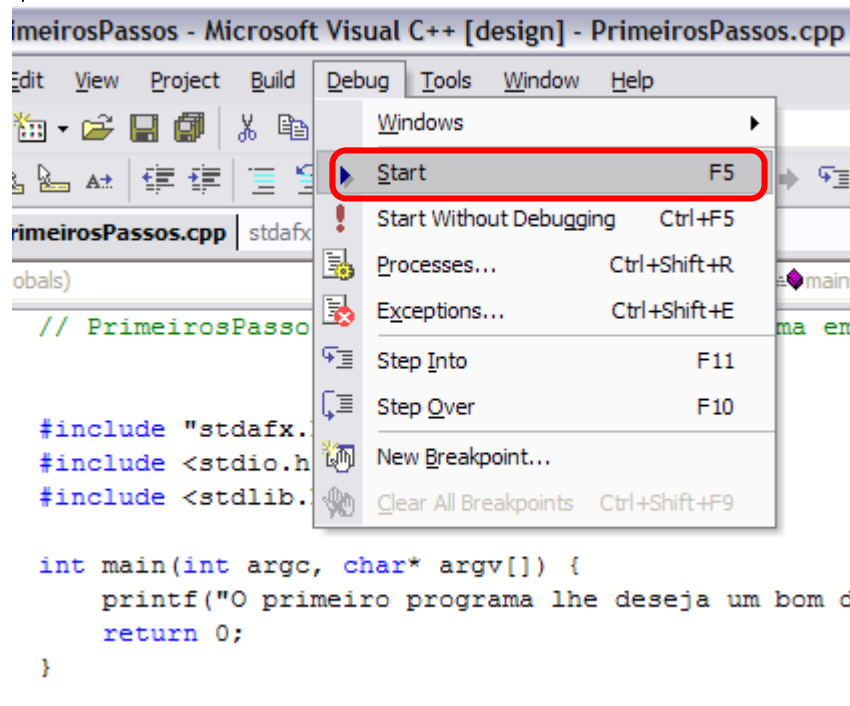
Primeiro Programa



```
ProgramaBasico - Microsoft Visual C++ [design] - main.c
File Edit View Project Build Debug Tools Window Help
main.c
(Globals) main
// PrimeirosPassos.cpp : Nosso primeiro programa em C
// Sempre incluir estas duas linhas.
// Elas consultam dois arquivos que contêm as definições dos com
// mais comuns de C.
#include <stdio.h>
#include <stdlib.h>
// Bloco principal de execução (função main).
// É aqui que começa a execução do programa.
// Sempre declare a função mais como na linha abaixo.
int main(int argc, char* argv[]) {
    // Imprime uma linha de texto
    printf("O primeiro programa lhe deseja um bom dia!\n");
    // O comando return finaliza o programa
    return 0;
}
Solution Explorer - Program...
Solution 'ProgramaBasico' (1 projec
ProgramaBasico
References
Source Files
main.c
Header Files
Resource Files
Task List
Ready Ln 1 Col 1 Ch 1 INS
```

Primeiro Programa

Executar:

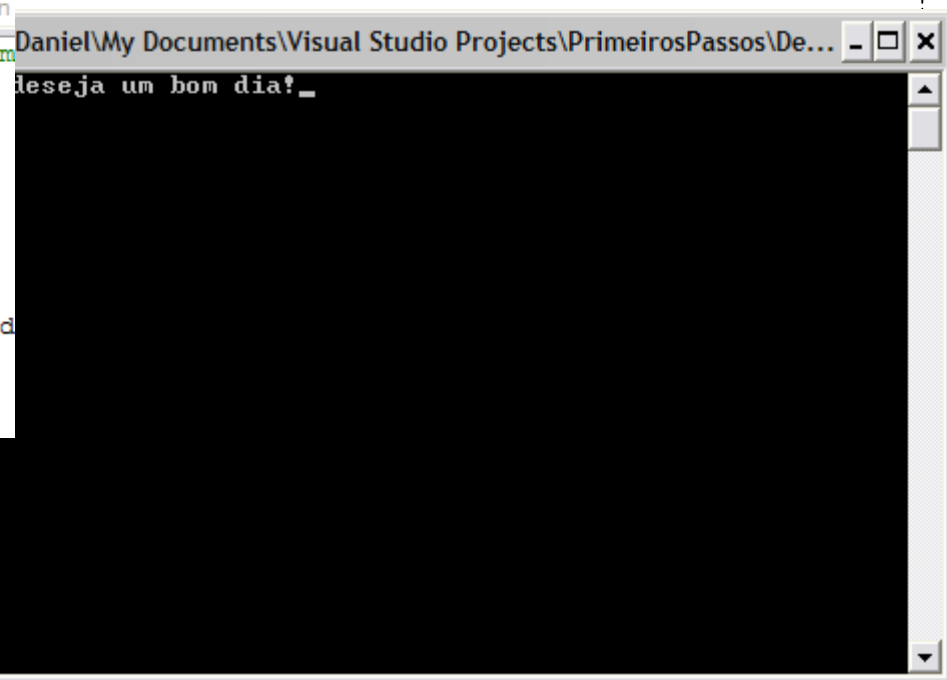


The screenshot shows the Microsoft Visual C++ IDE interface. The 'Debug' menu is open, and the 'Start' option is highlighted with a red rectangle. The 'Start' option is associated with the F5 key. Other options in the menu include 'Start Without Debugging', 'Processes...', 'Exceptions...', 'Step Into', 'Step Over', 'New Breakpoint...', and 'Clear All Breakpoints'. The background shows the source code for 'PrimeirosPassos.cpp'.

```
// PrimeirosPassos.cpp

#include "stdafx.h"
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char* argv[]) {
    printf("O primeiro programa lhe deseja um bom dia!\n");
    return 0;
}
```



The screenshot shows a terminal window with the output of the program. The text displayed is "O primeiro programa lhe deseja um bom dia!_" followed by a cursor. The window title is "Daniel\My Documents\Visual Studio Projects\PrimeirosPassos\De...".

```
O primeiro programa lhe deseja um bom dia!_
```