

**Edwar Saliba Júnior**

**Depuração de Código Utilizando o Dev-C++  
- Dicas -**

Versão 1.0

**Belo Horizonte  
Junho de 2009**

## Sumário

Dicas .....	2
Configuração do Dev-C++ .....	2
Como depurar um código no Dev-C++ .....	5
Referências Bibliográficas.....	9

## Dicas

Para fazermos a depuração de um código escrito em linguagem C/C++ utilizando o ambiente Dev-C++, faz-se necessário configurar alguns parâmetros para que o ambiente de depuração funcione.

## Configuração do Dev-C++

Para configurar o ambiente Dev-C++ para depuração, vá ao *menu Tools* (Ferramentas) | *Compiler Options* (Opções do Compilador). Conforme fig. 01.

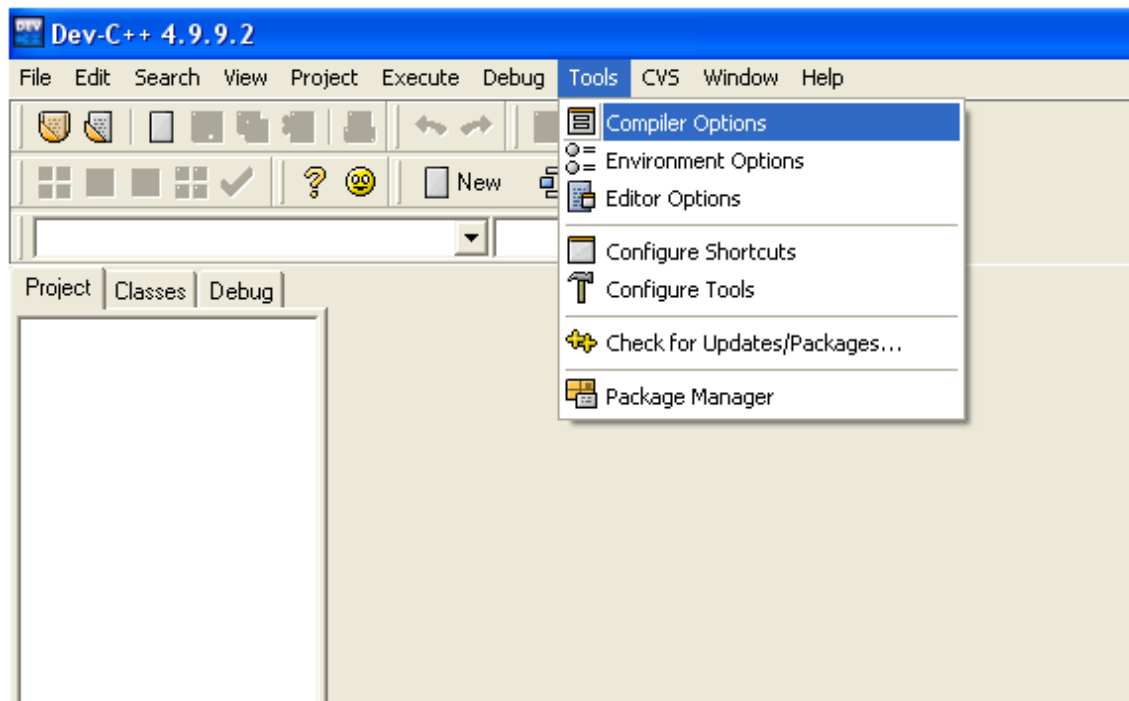


Fig. 01 - Menu Principal

Ao clicar na opção *Compiler Options*, uma nova tela será aberta (fig. 02). Nesta tela existe um *Check Box* com o nome de *Add the following commands when calling compiler:* (Retângulo vermelho na fig. 02), marque-o.

No campo logo abaixo do *Check Box* mencionado anteriormente (Seta verde da fig. 02), adicione a seguinte diretiva de compilação:

`-O0` (sinal de subtração, letra "O" maiúscula, zero)

Sua tela deverá ficar semelhante à mostrada na fig. 03.

Feito isto, clique na aba *Settings* (Retângulo Azul, Fig. 02).

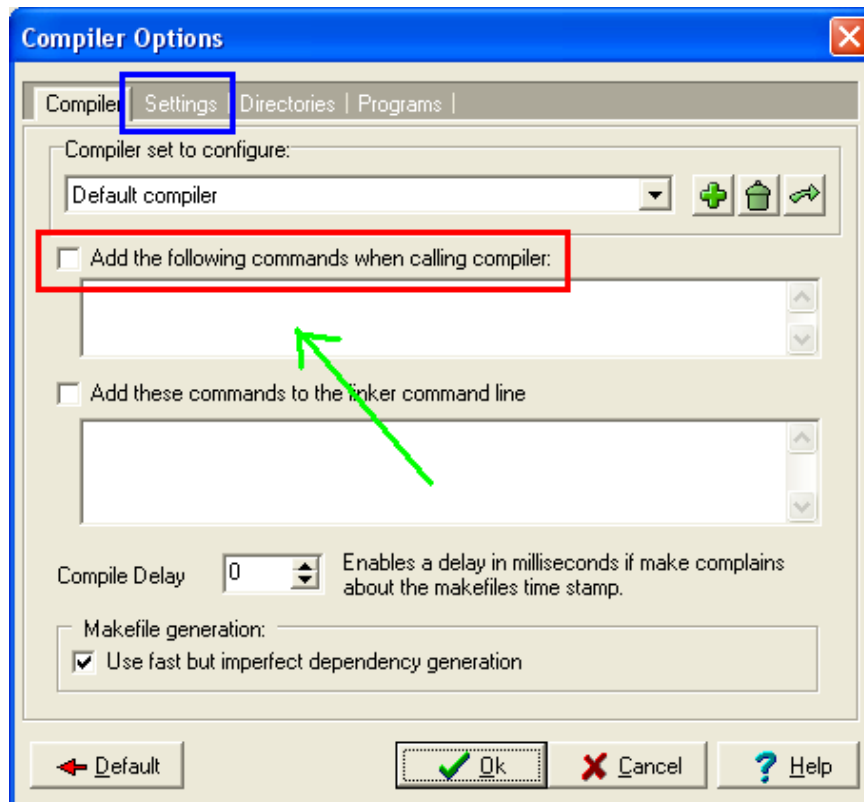


Fig. 02 - Opções do Compilador

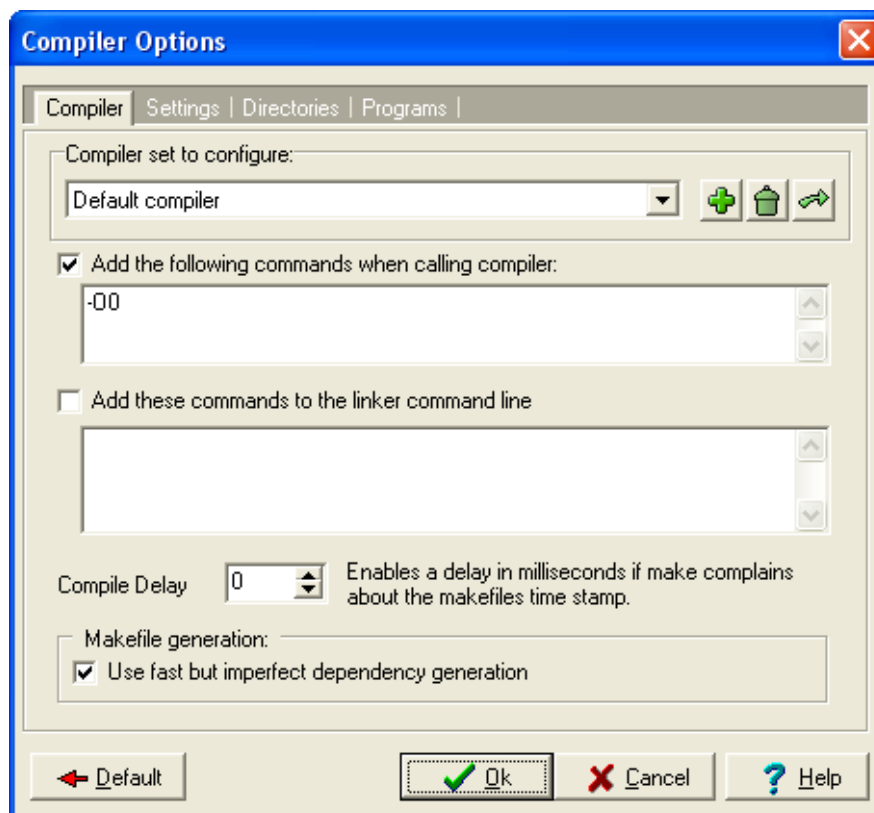


Fig. 03 - Diretiva de Depuração

Sua tela estará semelhante à apresentada na fig. 04.

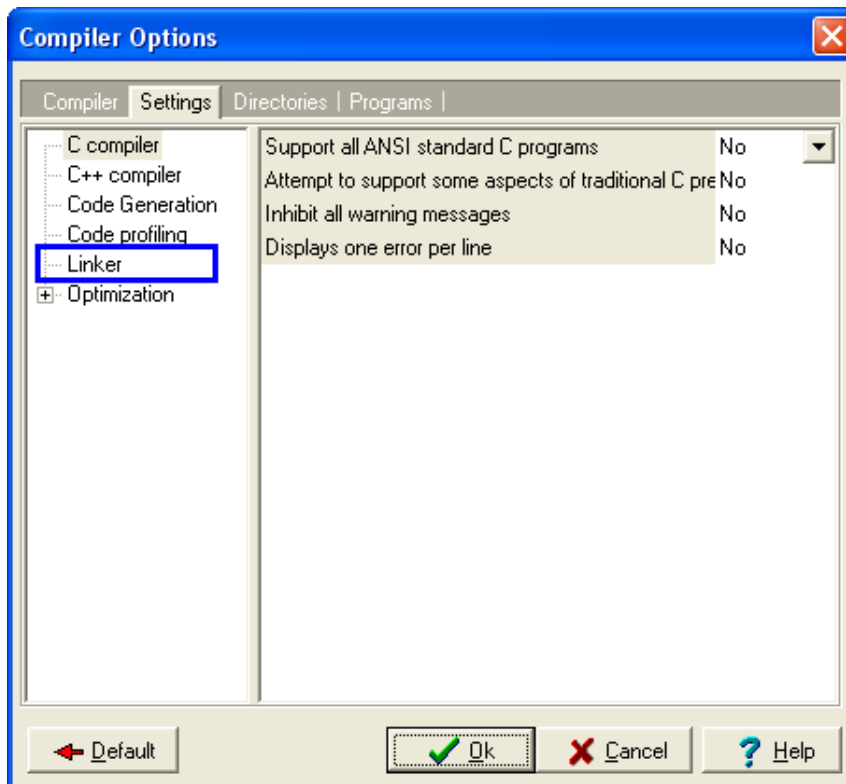


Fig. 04 - Aba Settings

Nas opções apresentadas no lado esquerdo, clique no item *Linker* (Retângulo Azul da fig. 04). As opções do lado direito mudarão para as opções apresentadas na fig. 05.

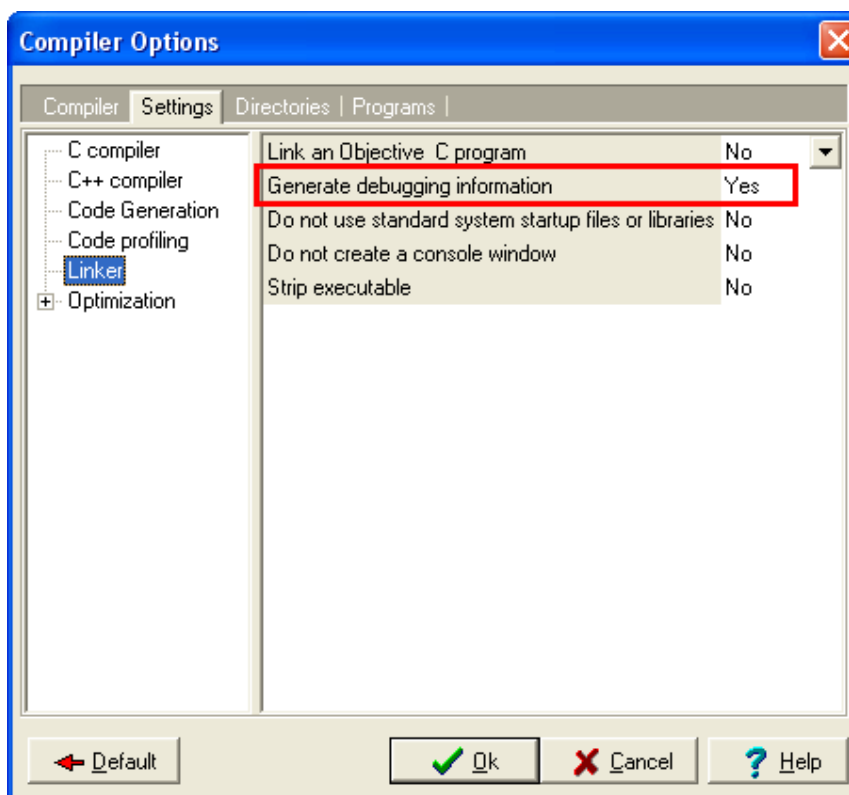


Fig. 05 - Linker

Certifique-se que a opção *Generate debugging information* esteja parametrizada com *Yes* (Retângulo Vermelho na fig. 05).

Feito isto, é só clicar no botão *Ok*. Sua configuração para depuração já está pronta.

## Como depurar um código no Dev-C++

Para depurar um código qualquer, você precisa abrir o referido código e marcar os devidos *breakpoints* (Pontos de parada).

Para tanto é só clicar na coluna negra no meio do ambiente Dev-C++ (fig. 06), na altura da linha onde se pretende interromper a execução do código.

Imediatamente aparecerá uma faixa vermelha sobre a linha escolhida, evidenciando o *breakpoint*. Aparecerá também uma bolinha vermelha com um "v"zinho verde-limão (ao meio), na coluna negra onde se clicou. (Veja fig. 06)

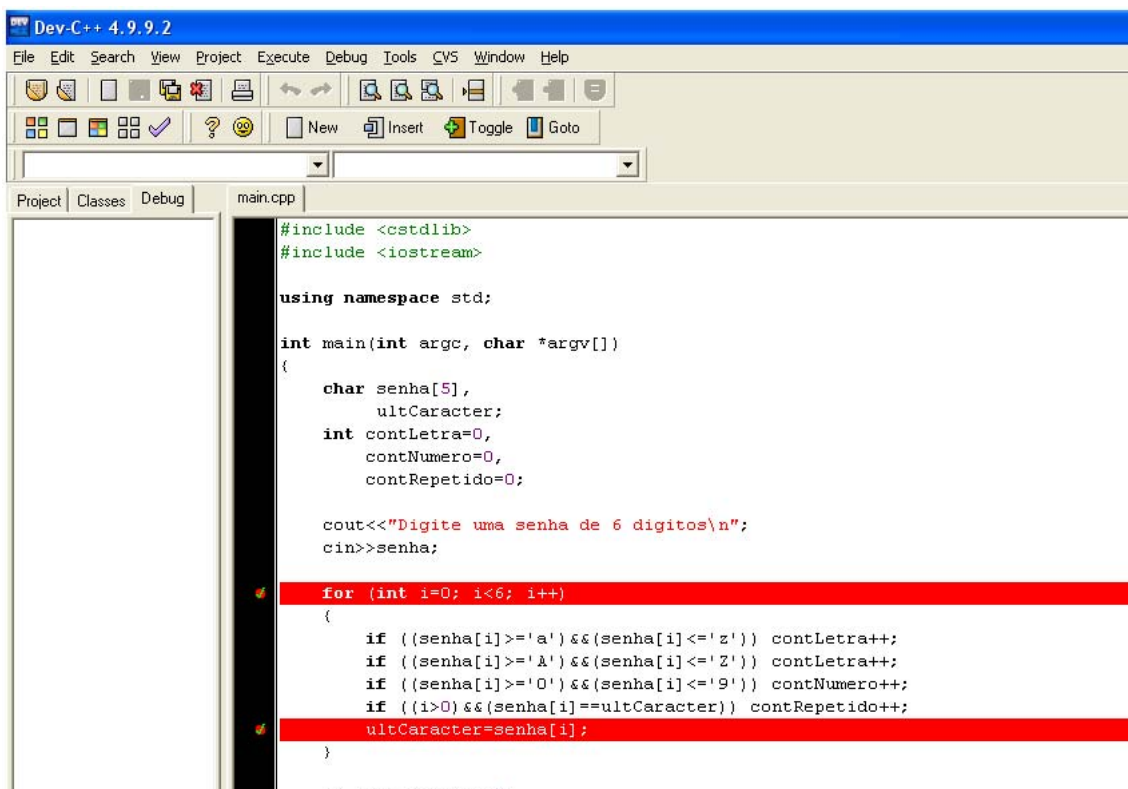


Fig. 06 - Breakpoints

Para executar o código em modo de depuração, basta apertar a tecla F8 ou escolher a opção *Debug* no *menu Debug* (fig. 07).

Na fig. 07 podemos notar o que *menu Debug* possui outras opções. Estas outras opções são fundamentais na depuração de um código.

São elas:

- *Debug* - Esta opção executa o código em modo de depuração.
- *Stop Execution* - Para a depuração.
- *Parameters...* - Utilizado para adição de parâmetros de depuração ou de otimização.
- *Toggle breakpoint* - Utilizada para marcar *breakpoints* em um código qualquer.
- *Next Step* - Executa o código linha a linha.
- *Step Into* - Entra em funções (quando possível).
- *Continue* - Executa o código diretamente até encontrar um *breakpoint*.
- *Run to Cursor* - Executa seu programa até encontrar o primeiro *breakpoint*.
- *Add Watch* - Opção utilizada para visualização dos valores de objetos e variáveis do programa. Para tanto, basta apertar a tecla F4 e digitar, na caixa de diálogo que aparecerá, o nome da variável que será observada. Ao adicionar uma variável para observação ela aparecerá no lado esquerdo da interface do Dev-C++, na aba *Debug* (Retângulo vermelho da fig. 08).
- *Watch Variables* - Atalho para visualizar a aba *Debug* (Retângulo vermelho da fig. 07), caso esta ainda não esteja visível.
- *View CPU Window* - Mostra o que está acontecendo no processador de seu computador. Mostra os comandos que serão executados em linguagem *Assembly*, e o que está armazenado nos registradores do processador no momento da execução (Fig. 09).

Obs.: Algumas opções só estarão ativas em modo de depuração.

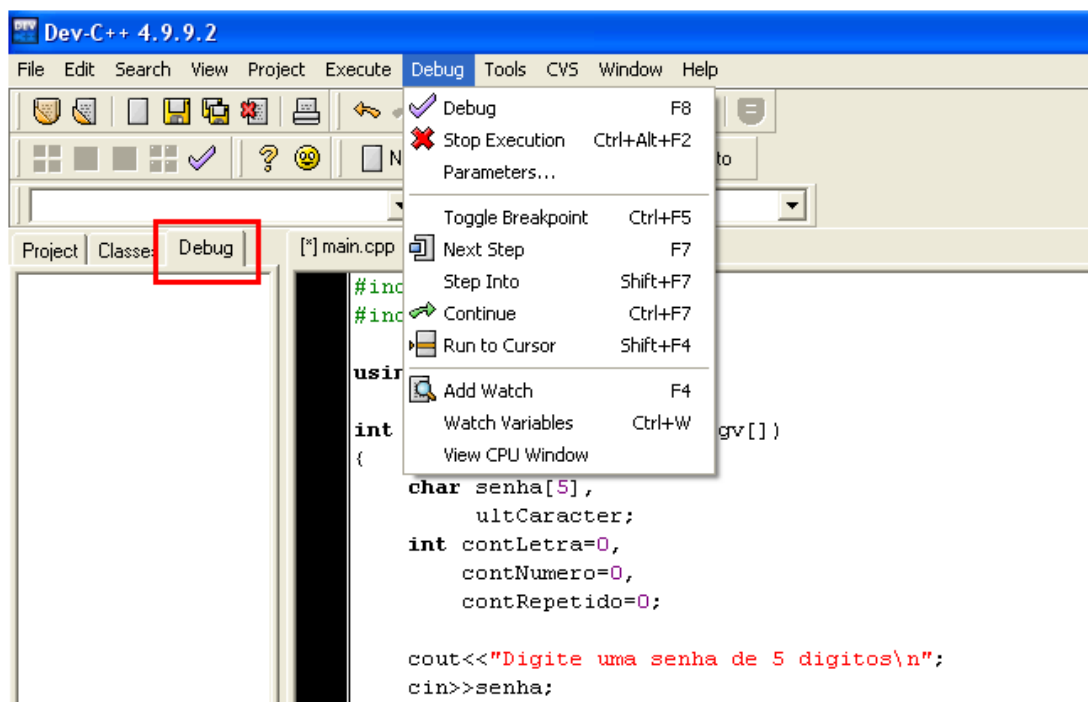


Fig. 07 - Menu de Depuração

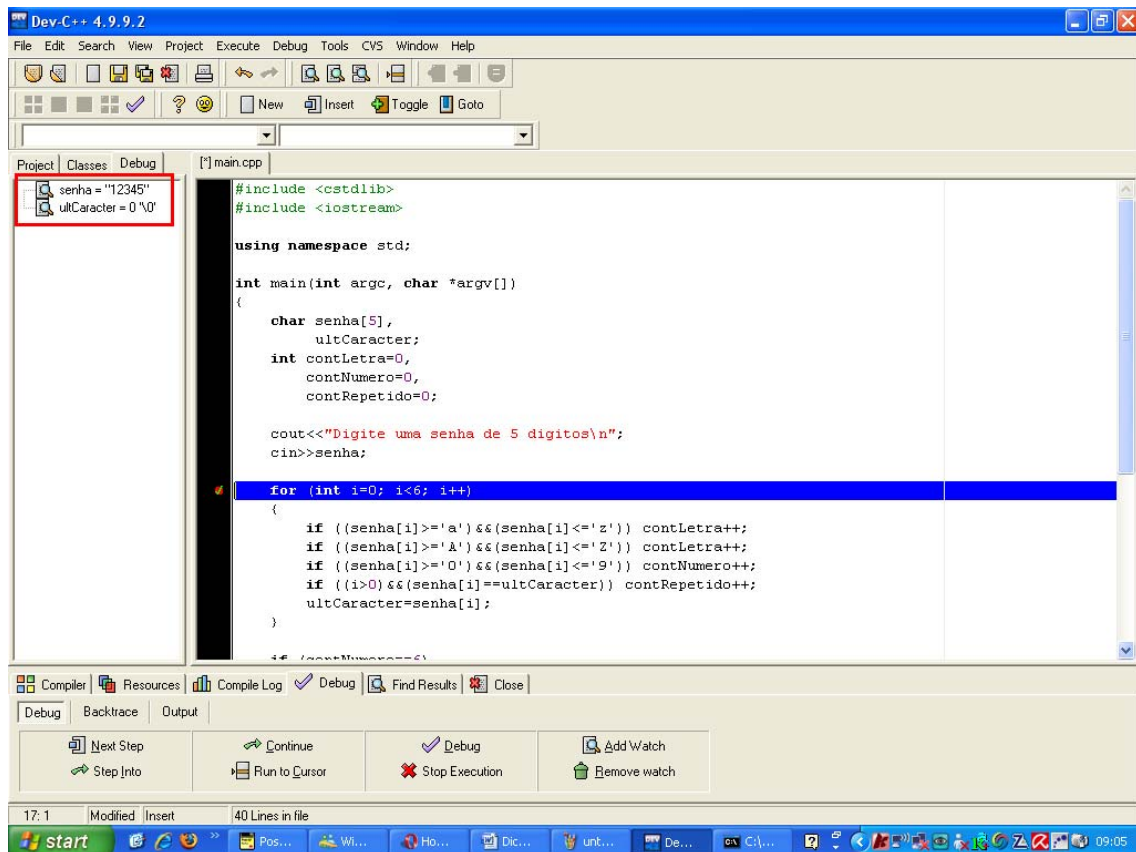


Fig. 08 - Watch

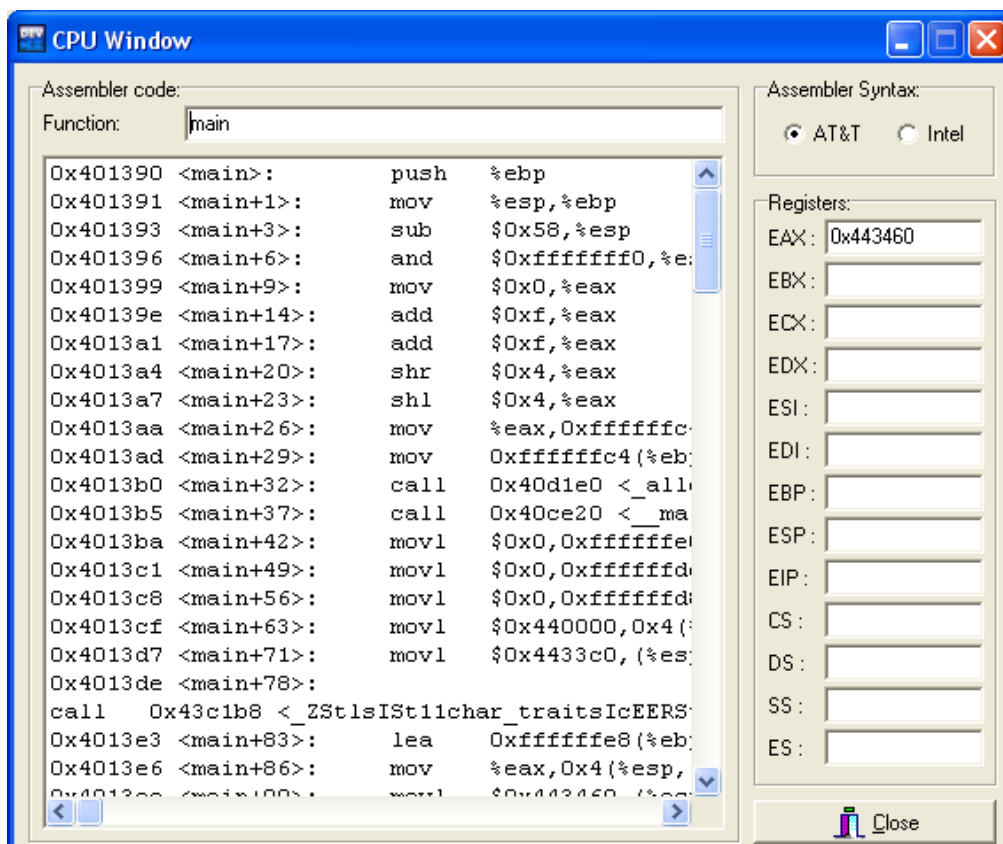


Fig. 09 - CPU Window



Agora que já sabemos para que serve cada opção do *menu Debug*, podemos começar a depuração de nosso *software*.

Lembre-se: para começar a depuração é necessário que o código a ser depurado esteja compilado. Para compilar o código basta utilizar as teclas de atalho `Ctrl + F9` ou a opção de compilação no *menu Execute | Compile*.

Se você já marcou seus *breakpoints* no código-fonte, aperte a tecla `F8` ou vá ao *menu Debug | Debug* para execução do código em modo de depuração.

O programa será executado normalmente até que seja encontrado ou um *breakpoint* ou um comando de leitura de dados (Ex.: `cin`, `scanf` e etc.).

Caso o programa execute um comando de leitura de dados a tela de entrada de dados será inicializada e o compilador aguardará até que um valor seja digitado e a tecla *Enter* seja pressionada (Fig. 10).

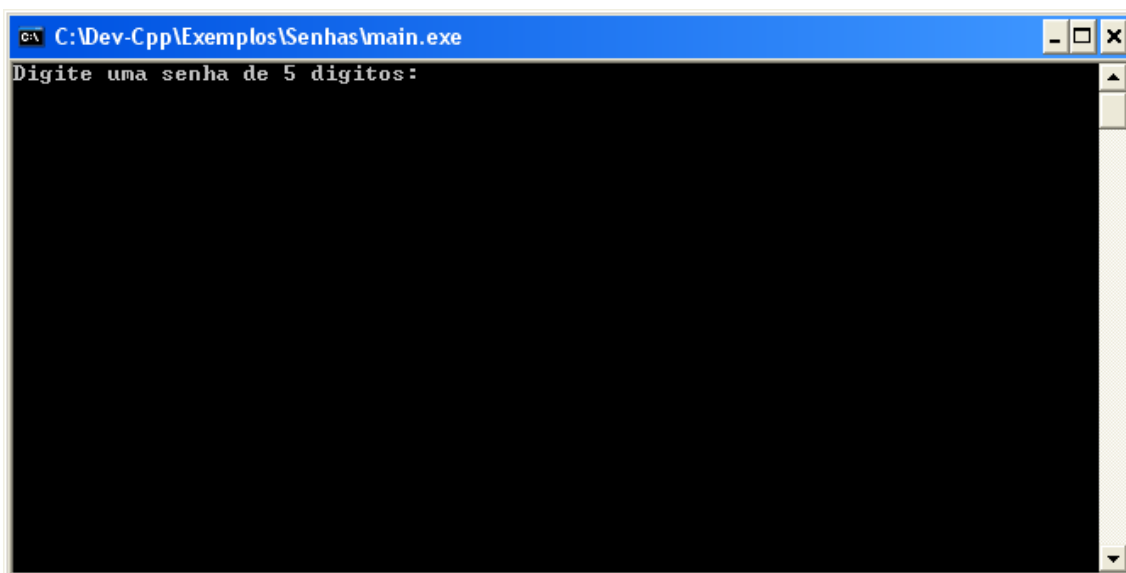


Fig. 10 - Comando de Entrada de Dados

Feito isto volte ao ambiente de depuração (Dev-C++) e continue a depuração do seu código.

Quando quiser observar o valor de alguma variável, adicione-a a tela de observação pressionando a tecla `F4` ou pelo *menu Debug | Add Watch*.

Para executar o código linha a linha pressione a tecla `F7` ou o *menu Debug | Next Step*.

Para entrar em funções (quando possível), na linha da função desejada, pressione o atalho `Shift + F7` ou pressione a opção *Step Into* no *menu Debug*. Assim que já estiver dentro da função, poderá executá-la linha a linha com a tecla `F7`.

## Referências Bibliográficas

ALVES, Maria Bernardete Martins; ARRUDA, Susana Margareth. **Como fazer referências**: bibliográficas, eletrônicas e demais formas de documento. Disponível em:  
<[www.nfr.ufsc.br/pen/docs/Modulo1Referencias.pdf](http://www.nfr.ufsc.br/pen/docs/Modulo1Referencias.pdf)> Acesso em: 12 fev. 2009.

HOW do I debug using Dev-C++?. [S.I. : s.n.]. Disponível em:  
<[http://eilat.sci.brooklyn.cuny.edu/cis1\\_5/HowToDebug.htm](http://eilat.sci.brooklyn.cuny.edu/cis1_5/HowToDebug.htm)>  
Acesso em: 04 jun. 2009.