



Universidade Federal de Minas Gerais

Escola de Engenharia

Departamento de Engenharia Eletrônica

Disciplina: Automação em Tempo Real (ELT012)

Data: 17/04/2008

Professor: Luiz T. S. Mendes

Guia prático para conversão de programas do Visual C++ 6.0 para o Visual C++ 2008 Express Edition

Este guia destina-se a orientar como converter os projetos disponibilizados na página *web* do livro “Programação Concorrente em Ambiente Windows: Uma Visão de Automação”, originalmente elaborados para a ferramenta *Microsoft Visual C++ 6.0* (MS-VC++6.0), para o ambiente da ferramenta *Visual C++ 2008 Express Edition*.

Antes de proceder ao processo de conversão, certifique-se que a ferramenta *Visual C++ 2008 Express Edition* foi corretamente instalada em seu sistema, e que as pastas referentes aos projetos originais em MS-VC++6.0 foram adequadamente reconstruídas em diretórios apropriados, incluindo a pasta de nome “Include” que contém os arquivos de cabeçalhos (*header files*) que são referenciados pelos programas-fonte de cada projeto.

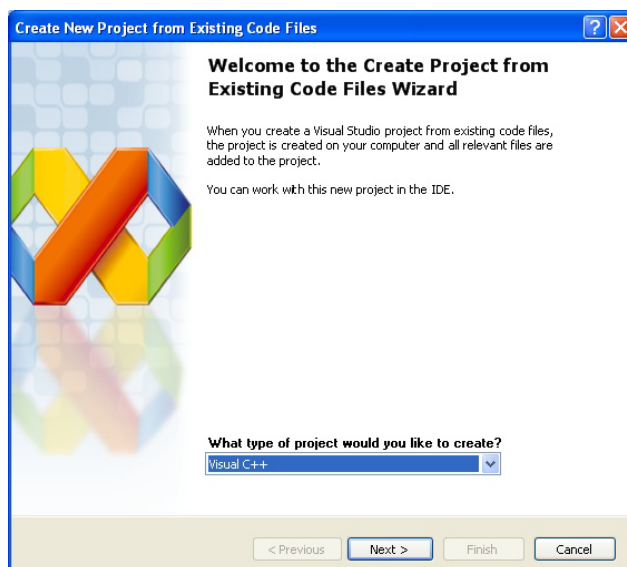
O processo de conversão será exemplificado para o projeto referente ao programa 4.4 (*Prog44WaitableTimers*), último visto na aula do dia 11/04/2008.

Existem duas formas de conversão de projetos para o *Visual C++ 2008 Express Edition*: a primeira utiliza o *Create New Project From Existing Code Files Wizard*, recurso que automatiza grande parte do processo e, por isto, é o mais simples; a segunda forma usa a criação de um projeto vazio ao qual adicionam-se os arquivos necessários. Neste guia, somente a primeira forma será apresentada.

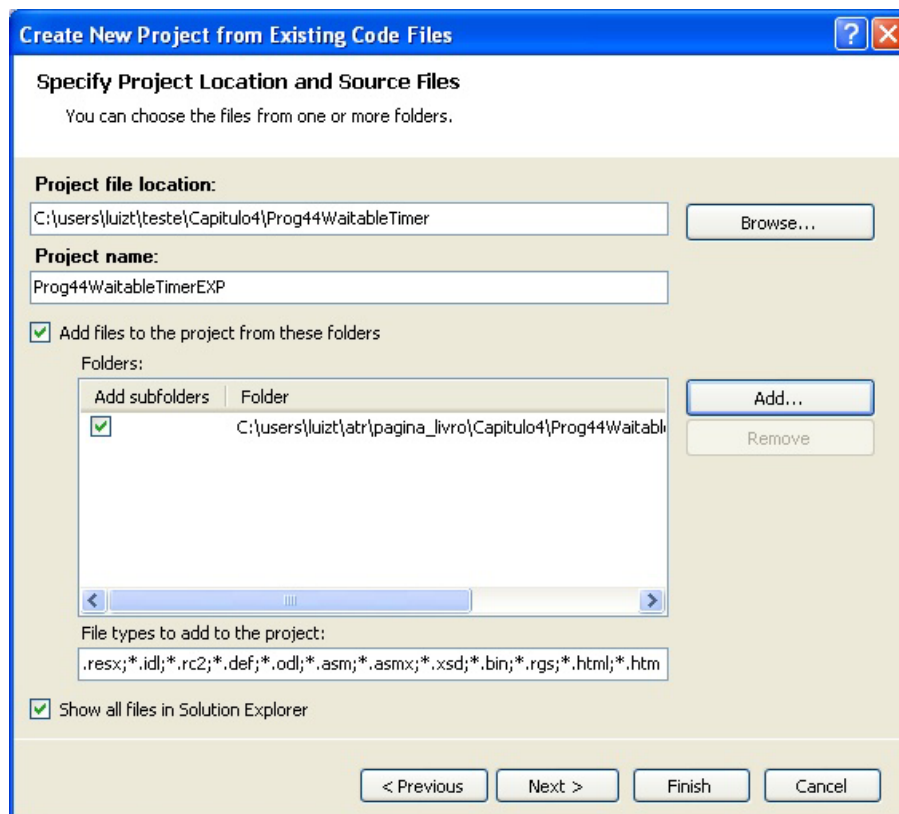
Os exemplos apresentados a seguir correspondem à versão na língua inglesa do *Visual C++ 2008 Express Edition*, mas naturalmente tais janelas terão correspondência exata com as equivalentes da versão em português.

1. Conversão através do “Create New Project From Existing Code Files Wizard”

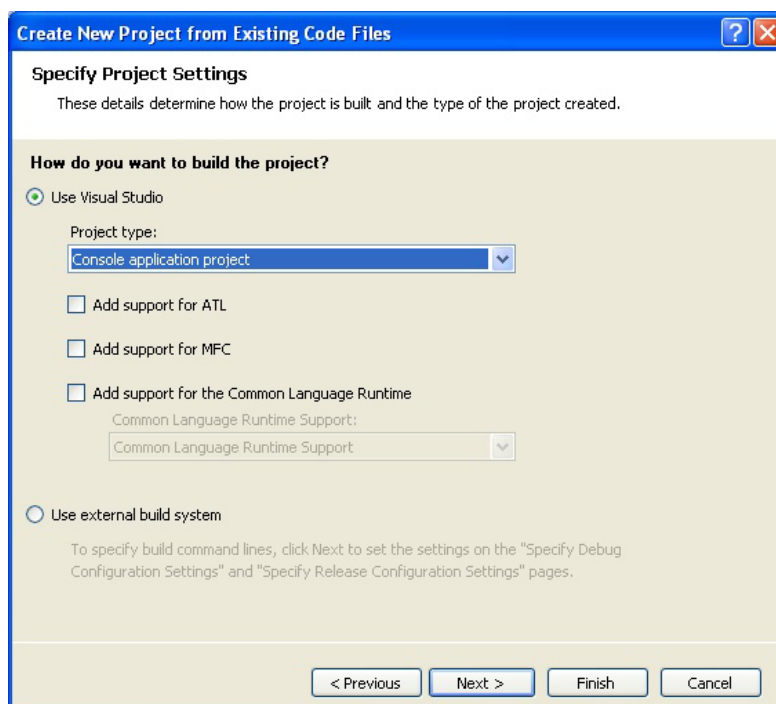
Execute o *Visual C++ 2008 Express Edition* e, neste, selecione *File* → *New* → *Project From Existing Code*. A janela abaixo será mostrada:



Nesta janela, selecione obviamente a opção “Visual C++” caso você tenha mais de uma ferramenta do Visual Studio 2008 Express instalada, e clique em *Next*. Nova janela será então apresentada:

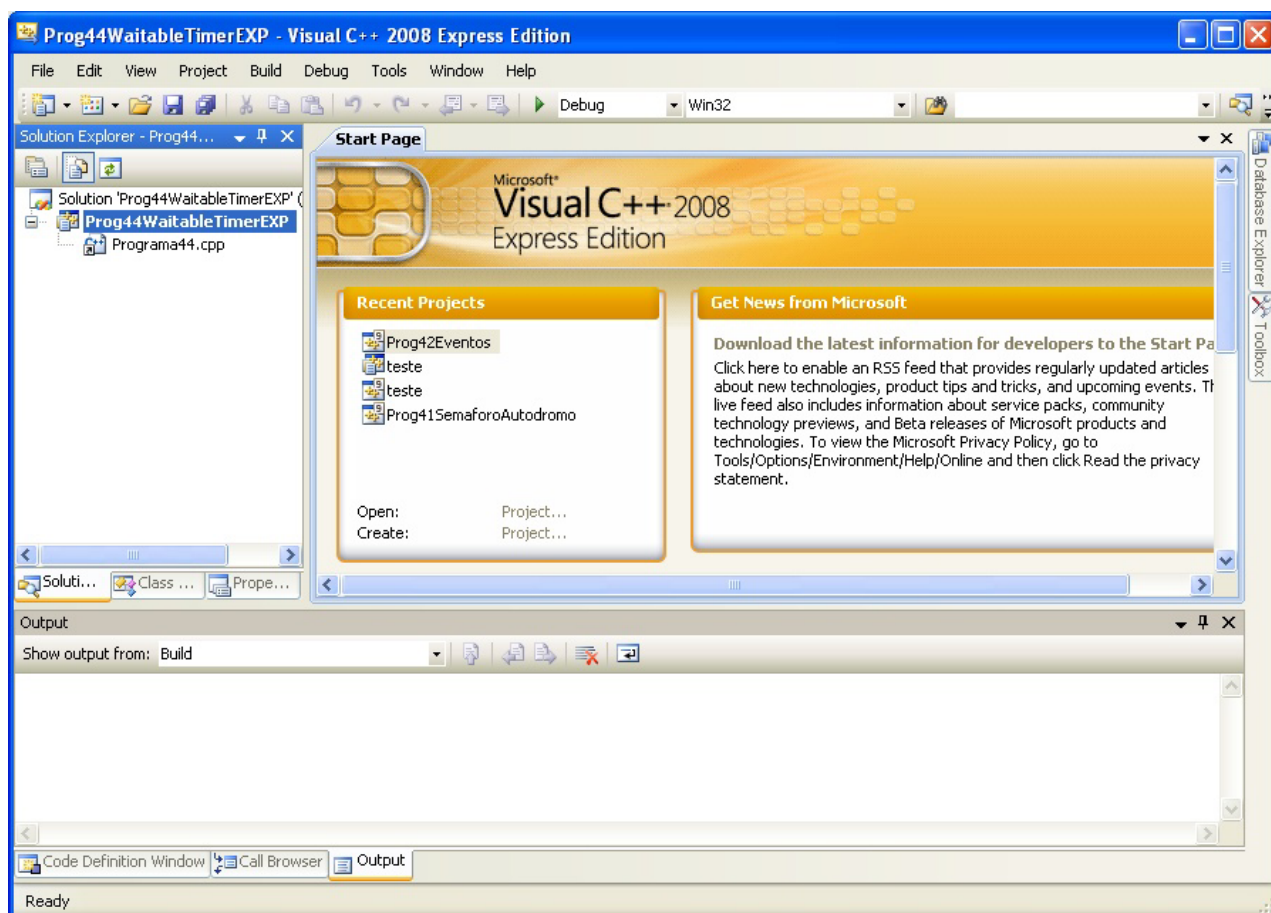


Nesta janela, indique o diretório onde o projeto convertido será salvo, no campo “*Project file location*”; o nome do projeto convertido, no campo “*Project name*”; e clique no botão “Add” para especificar o diretório correspondente ao projeto original a ser convertido, em nosso caso *Prog44WaitableTimers*. É possível especificar o mesmo diretório tanto para o projeto convertido quanto para o diretório que contém o projeto a converter, mas neste caso os arquivos originais deste último serão destruídos no processo de conversão. Clique em “*Next*” (**Atenção!** Não clique em “*Finish*” ainda neste ponto, pois do contrário será criada uma aplicação gráfica por *default*, e não uma aplicação em modo console como desejamos). Nova janela será apresentada:



Como mostrado na mesma, selecione a opção “*Use Visual Studio*” e selecione a opção de tipo de projeto correspondente a “*Console application project*”. Em seguida, clique finalmente em “*Finish*”. O projeto convertido será criado no diretório anteriormente especificado; consulte este último com o *Windows Explorer* para certificar-se que os arquivos referentes ao projeto foram realmente criados.

Como resultado da conversão, o *Visual C++ 2008 Express Edition* criará uma “Solução” (*Solution*), que essencialmente é um objeto que corresponde à agregação de diversos projetos, imagens, e outros arquivos necessários à execução de uma aplicação completa. No nosso caso, obviamente, o único componente da “Solução” criada será o nosso projeto convertido. A tela principal do *Visual C++ 2008 Express Edition* terá então um aspecto semelhante à seguinte imagem:

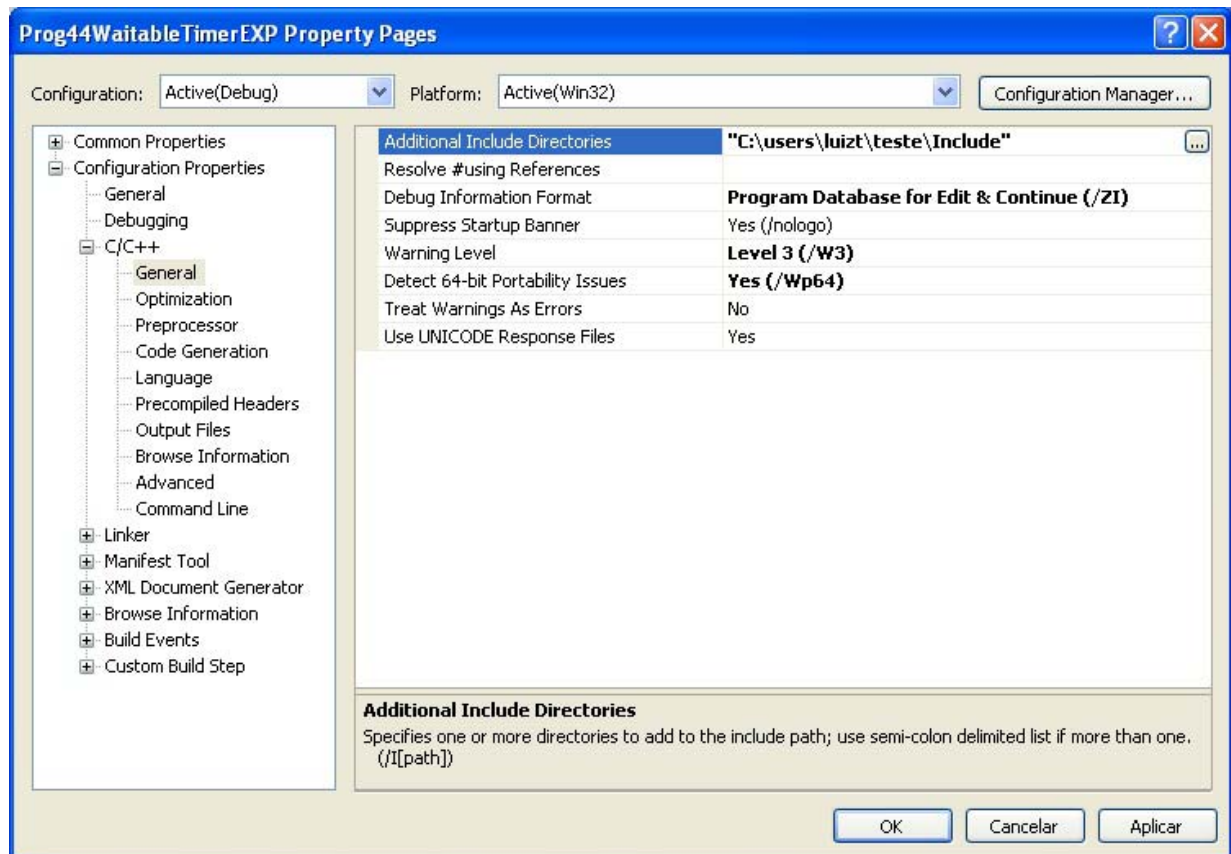


Observe, a esta altura, que neste exemplo o nome do novo projeto convertido foi escolhido como *Prog44WaitableTimersEXP* para diferenciá-lo do nome do programa original a ser convertido. Clique no item “Programa44.cpp”, na árvore da “Solução” contida no lado esquerdo da janela, para visualizar o programa-fonte correspondente ao projeto *Prog44WaitableTimersEXP*.

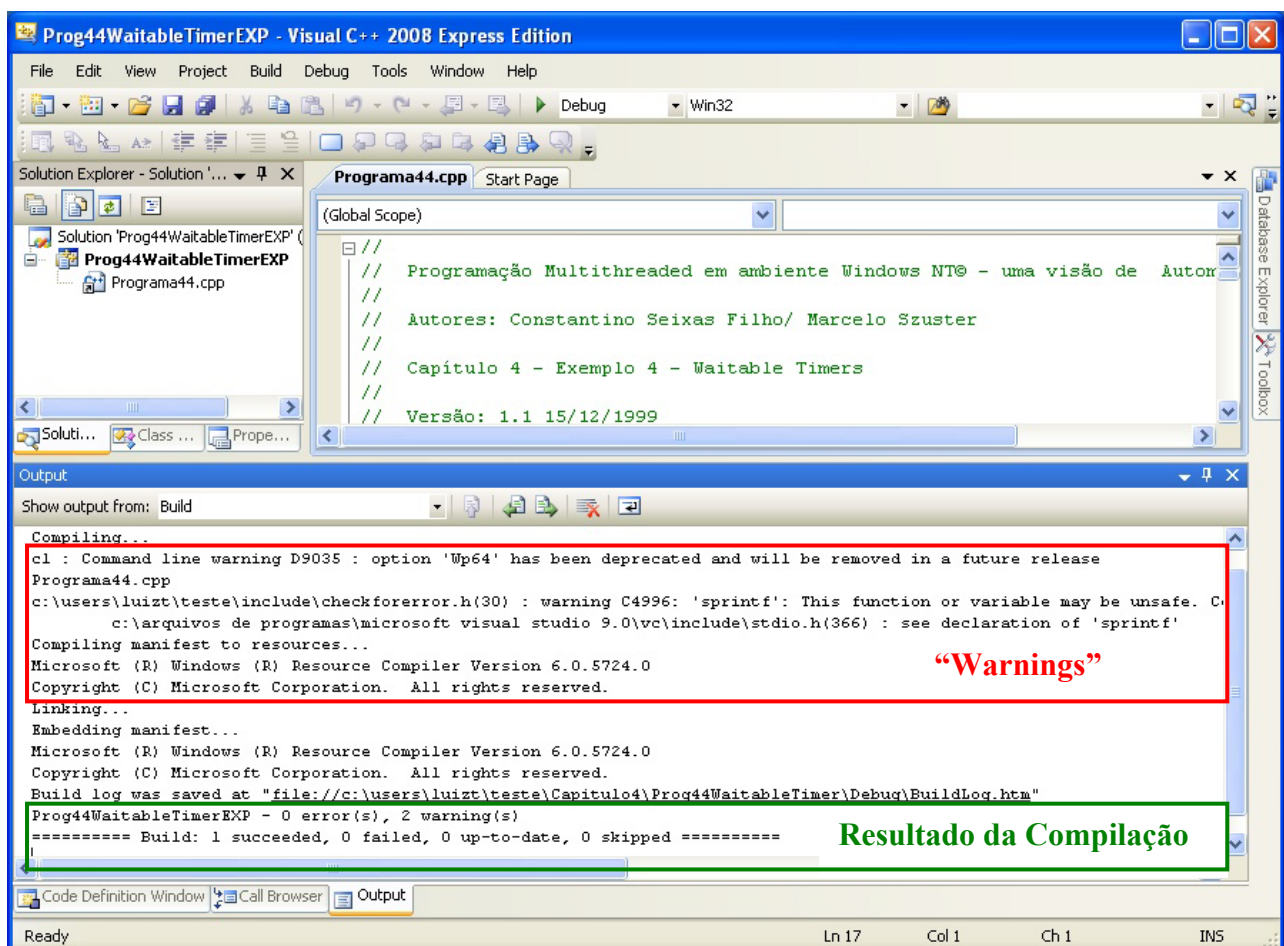
Antes de compilar o programa, é necessário especificar a localização do diretório que contém os arquivos de cabeçalho (*header files*) referenciados pelo programa-fonte, tal como “*bGetFloat.h*” e “*CheckForError.h*”. Para tal, siga os passos:

- Selecione *Project* → *Prog44WaitableTimersEXP Properties*
- Na janela de propriedades, selecione, na árvore de atributos à esquerda da mesma, a opção *Configuration Properties* → *C/C++* → *General*
- No lado direito da janela, especifique o diretório contendo os arquivos de cabeçalho através do campo “*Additional Include Directories*”.
- Clique em “OK”.

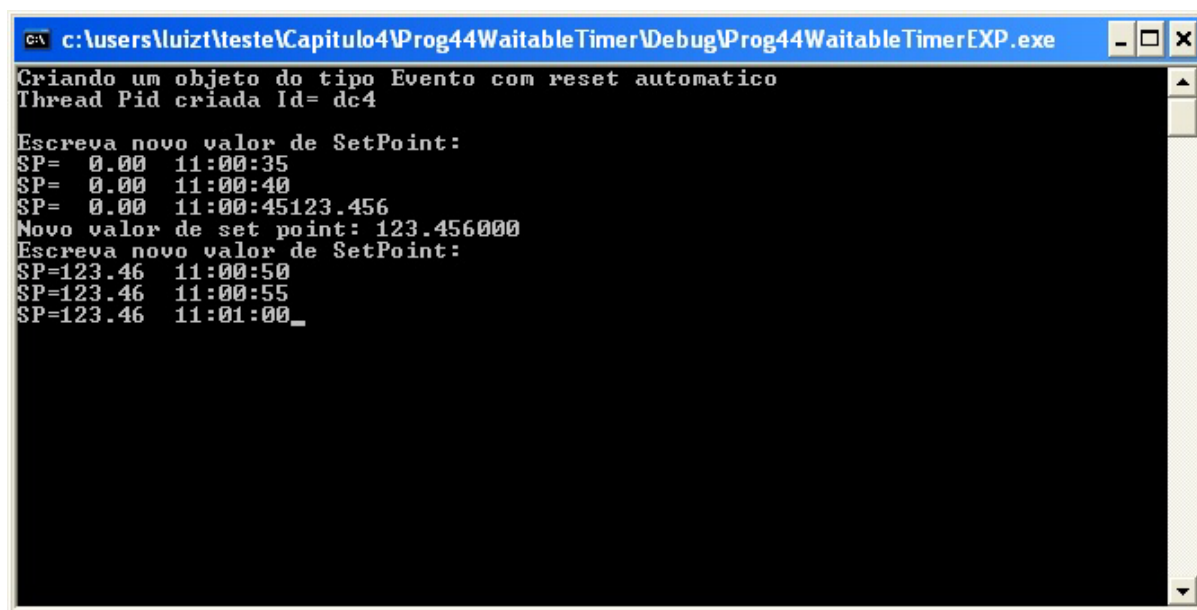
A imagem na próxima página exemplifica o processo descrito acima:



Pronto! O projeto convertido está apto a ser compilado. Para tal, selecione agora *Build* → *Build Prog44WaitableTimersEX*. A compilação será disparada, e o resultado da mesma apresentado na parte inferior da tela do *Visual C++ 2008 Express Edition*



Como se pode observar, a compilação foi bem-sucedida, ocorrendo apenas 2 “warnings” que entretanto não interferem no sucesso da mesma. Para executar o programa gerado, selecione agora *Debug* → *Start Debugging*:



Ecco ! Tudo funcionou como devia. O mesmo processo pode então ser repetido para cada um dos projetos originais referentes ao livro-texto.

2. Compreensão e supressão das mensagens de “warning” na compilação

Mencionamos anteriormente que os “warnings” ocorridos na compilação do programa não interferem no sucesso da compilação, mas em geral os mesmos sinalizam aspectos importantes que devem ser corretamente entendidos para que nos certifiquemos de que a aplicação construída com o *Visual C++ 2008 Express Edition* executará corretamente. Observando as mensagens de “warning” ocorridas na compilação do projeto *Prog44WaitableTimersEXP*, vemos os seguintes textos:

```
cl : Command line warning D9035 : option 'Wp64' has been deprecated and will
be removed in a future release
```

```
Programa44.cpp
c:\users\luizt\teste\include\checkforerror.h(30) : warning C4996: 'sprintf':
This function or variable may be unsafe. Consider using sprintf_s instead. To
disable deprecation, use _CRT_SECURE_NO_WARNINGS. See online help for details.
c:\arquivos de programas\microsoft visual studio 9.0\vc\include\stdio.h(366):
see declaration of 'sprintf'
```

A primeira mensagem diz respeito à opção de compilação “*Wp64*”, que executa uma verificação de compatibilidade do programa executável quanto a CPUs de 64 bits. A mensagem diz que esta opção deve ser evitada e será removida em versões futuras. Ou seja, trata-se de uma mensagem inócua para nós; além do mais, em geral executamos nossas aplicações apenas em computadores com CPUs de 32 bits. Para remover este “warning”, selecione novamente *Project* → *Prog44WaitableTimersEXP Properties*; na janela de propriedades, selecione então *Configuration Properties* → *C/C++* → *General* e então defina a opção “*Detect 64-bit portability issues*” como “*No*”.

A segunda mensagem é mais importante, apesar de não interferir muito na aplicação referente ao exercício *Prog44WaitableTimers*. Ocorre que as funções clássicas de tratamento de *strings* em C/C++ como `sprintf()`, `strcpy()`, etc. são consideradas inseguras pela possibilidade de, por erro ou propositalmente por parte do programador, poderem sobrescrever indevidamente áreas de

memória do computador, causando, assim, efeitos indesejados que podem ir desde a mera execução incorreta da aplicação até um eventual comprometimento da segurança de acesso do computador. No ambiente do *Visual C++ 2008 Express Edition*, temos duas opções: (1) simplesmente inibir a mensagem de erro; ou (2) substituir estas funções por outras equivalentes, porém mais seguras, como `sprintf_s()`, `strcpy_s()` e etc. No primeiro caso, basta incluir a diretiva

```
#define _CRT_SECURE_NO_WARNING
```

entre as declarações usuais no início do programa. No segundo caso, pode ser feita a substituição manual das funções originais de manipulação de strings pelas equivalentes seguras, mas isto requer um certo esforço pelo fato destas últimas requererem, em geral, um parâmetro a mais que é o tamanho do *string* a ser manipulado. Felizmente, pode-se deixar a cargo do *Visual C++ 2008 Express Edition* esta substituição automática, bastando para isto incluir a diretiva

```
#define _CRT_SECURE_CPP_OVERLOAD_STANDARD_NAMES 1
```

no início do programa, antes da primeira diretiva `#include` (para evitar outros “warnings” decorrentes da re-definição deste `#define`)..

Com estes ajustes, pode-se ter então uma compilação livre dos “warnings” apresentados:

