



**Нижегородский государственный университет  
им. Н.И.Лобачевского**

***Факультет Вычислительной математики и кибернетики***

**Зимняя школа 2007**

***Инструменты параллельного программирования для  
систем с общей памятью***

**Введение в OpenMP. Практика 1**

Сысоев А.В.  
Кафедра математического  
обеспечения ЭВМ

# Содержание

---

## Упражнения:

- ❑ Настройка проекта в средах разработки семейства Microsoft Visual Studio
- ❑ Hello World!
- ❑ Скалярное произведение векторов



# Упражнение 1: Настройка проекта в средах разработки семейства Microsoft Visual Studio...

## □ MS VS 6.0

- Поддержка только в компиляторах Intel
- Компилятор Intel встраивается в среду разработки
- Включение компилятора Intel
  - Команда меню Tools->Intel® C++ Compiler Selection Tool
- Включение OpenMP
  - Настройки проекта: команда меню Project->Settings
  - Вкладка C\C++
  - В поле Project Options добавить ключ компилятора /Qopenmp
  - В файлах программы, использующих конструкции OpenMP подключить заголовочный файл <omp.h>



# Упражнение 1: Настройка проекта в средах разработки семейства Microsoft Visual Studio...

## □ MS VS .NET 2003

- Поддержка только в компиляторах Intel
- Компилятор Intel встраивается в среду разработки
- Включение компилятора Intel
  - Команда контекстного меню решения (solution)  
Convert to use Intel® C++ Project System
- Включение OpenMP
  - Настройки проекта: команда меню Project->Properties
  - В пункте C\C++->Language->Process OpenMP Directives выбрать “Generate Parallel Code (/Qopenmp)”
  - В файлах программы, использующих конструкции OpenMP подключить заголовочный файл <omp.h>



# Упражнение 1: Настройка проекта в средах разработки семейства Microsoft Visual Studio

## □ MS VS .NET 2005

- Поддержка в компиляторах Microsoft и Intel
- Компилятор Intel встраивается в среду разработки (начиная с версии 9.1)
- Включение компилятора Intel
  - Команда контекстного меню решения (solution)  
Convert to use Intel® C++ Project System
- Включение OpenMP
  - Настройки проекта: команда меню Project->Properties
  - В пункте C\C++->Language->OpenMP Support выбрать “Generate Parallel Code (/Qopenmp)”
  - В файлах программы, использующих конструкции OpenMP подключить заголовочный файл <omp.h>



# Упражнение 1.1: Hello World!

- ❑ Постановка задачи
  - Написать OpenMP версию программы «Hello World!»
  - Версия 1
    - Каждый поток определяет свой номер и выводит его на экран
- ❑ При работе в MS VS 6.0 и 2003 убедитесь, что вывод на экран неверен (на многоядерной/многопроцессорной системе)
  - Объясните, почему?



# Упражнение 1.1: Настройка используемых run-time библиотек в Microsoft Visual Studio

## ❑ MS VS .NET 6.0

- Настройки проекта: команда меню Project->Settings
- Вкладка C\C++
- В списке Use run-time library выберите необходимый способ генерации кода на основе multithreaded библиотек

## ❑ MS VS .NET 2003

- Настройки проекта: команда меню Project->Properties
- В пункте C\C++->Language->Code Generation->Runtime library выберите необходимый способ генерации кода на основе multithreaded библиотек



## Упражнение 2.1: Hello World!

- ❑ Постановка задачи
  - Написать OpenMP версию программы «Hello World!»
  - Версия 1
    - Каждый поток определяет свой номер и выводит его на экран
- ❑ Убедитесь, что вывод на экран неверен
  - Объясните, почему?
- ❑ Внесите исправления в настройки проекта
  - Убедитесь, что программа работает корректно





## Упражнение 2.2: Hello World!

---

### □ Постановка задачи

- Написать OpenMP версию программы «Hello World!»
- Версия 2
  - Каждый поток определяет свой номер и выводит его на экран
  - Дополнительно: поток 0 выводит общее число потоков



## Упражнение 2.3: Hello World!

### □ Постановка задачи

- Написать OpenMP версию программы «Hello World!»
- Версия 3
  - Каждый поток определяет свой номер и выводит его на экран
  - Дополнительно: поток 0 выводит общее число потоков
  - Дополнительно: номера потоков выводятся в порядке возрастания

### □ Объясните, имеет ли смысл добиваться жесткого порядка в реальных задачах?



# Упражнение 3: Скалярное произведение векторов

## □ Постановка задачи

- Написать OpenMP версию программы, вычисляющей скалярное произведение векторов заданного размера
- Вектора заполняются случайным образом
- Между потоками распределяется основной цикл расчета
- Замерить ускорение на различных размерах векторов
- Объяснить полученные результаты



# Вопросы

---

□ ???

