

3. Які недоліки мають 3DNow! команди?
4. Які недоліки мають SSE команди?
5. Вам треба обробляти дані типу __m256i. Яку перевірку необхідно обов'язково виконати перед використанням команд для обробки даних цих типів?
6. Як виділити пам'ять під масив довільної довжини таким чином, щоб можна було для цього масиву використовувати AVX команди?

4 ПОРІВНЯННЯ МЕТОДІВ СТВОРЕННЯ ПОТОКІВ В WINDOWS I OPEN MP

4.1 Мета роботи

При запуску програми операційна система створює один потік. Якщо виконується одна програма, то фактично використовується одне ядро процесору. Для забезпечення завантаження ядер необхідно, щоб одночасно необхідно було виконувати декілька потоків. Краще, якщо кількість готових потоків в кожен момент часу дорівнював числу ядер. Таким чином, проблема створення потоків, керуванням їх кількістю в залежності від кількості ядер, є актуальною задачею. Метою даної лабораторної роботи є порівняння методів створення потоків за допомогою потоків OS Windows і Open MP з боку витрат часу програміста на створення і додаткових витрат часу процесору.

4.2 Підготовка до роботи

4.2.1 Включення режиму Open MP

Створити проект⁴.

В закладці Project-><Ім'я проекту>Properties->Command Line в окні Additional Options додати ключ трансляції /openmp або в Project->< Ім'я проекту>Properties->Language встановити флажок OpenMP Support і натиснути кнопку Примінити, а потім ОК.

Перевірити, що режим використання OpenMP вмикнувся. При включенні цього режиму визначається змінна _OPENMP. Для перевірки наявності цієї змінної можна використовувати наступний код::

```
#include <stdio.h>
#include <omp.h>

int main() {
    #ifdef _OPENMP
        printf ("_OPENMP Defined\n");
    
```

⁴ Деякі версії компіляторів не працюють в разі створення Solution з декількома проектами

```

#else
    printf ("_OPENMP UnDefined\n");
#endif
}

```

В разі, коли режим Open MP включено, програма виведе _OPENMP Defined.

4.2.2 Визначення кількості ядер процесору

Для визначення кількості ядер можна використовувати 2 способи.

1. Без переключення в режим використання Open MP.
- 2 З переключенням

Визначення кількості ядер без переключення в режим використання Open MP

Для визначення використовуються функції операційних систем. Знайдіть ці функції і визначте кількість ядер.

Для визначення кількості ядер з переключенням використовується функція Open MP `omp_get_max_threads`, яка фактично повертає кількість ядер.

Порівняйте результати, отримані різними способами

4.2.3 Створення потоків.

Для створення потоків в Windows вивчіть відповідний розділ [1]. Для створення потоків в Open MP вивчіть директиви `parallel`. Якщо потоки створюються для паралельного виконання циклу, то необхідно використовувати директиву `for`.

Загальний вид директиви `parallel`:

```
#pragma omp parallel [clause[ [, ]clause] ...],
```

де *clause* – додаткові режими, які в найпростішому виді відсутні.

Загальний вид директиви `for`:

```
#pragma omp [parallel] for [clauses],
```

де *clause* – додаткові режими, які в найпростішому виді відсутні.

Обидві директиви можуть бути замінені однією:

```
#pragma omp parallel for
```

Цю директиву ми рекомендуємо використовувати для паралельного виконання циклу.

4.3 Порядок виконання лабораторної роботи

1. Включити режим Open MP і перевірити успішність його включення.
2. Визначити кількість потоків першим та другим способом. Порівняти отримані

значення.

3. Реалізувати послідовні функції, які обчислюють значення π (3 різні функції) з заданою точністю.
4. Визначити найкращу функцію з боку обчислювальної складності.
5. Визначити кількість ітерацій для отримання максимально можливої точності в разі використання даних типу double.
6. Реалізувати паралельно найкращу з обраних функцій. Створити за допомогою функції Windows потоки. Обрати кількість ітерацій як для послідовного варіанту.
7. Зробити висновки по ефективності використання Windows потоків по часу та точності.
8. Забезпечити паралельне виконання за допомогою директив Open MP. Порівняти результати з попередніми. Зробити висновки по ефективності використання Open MP потоків.
9. Визначте значення прискорення для паралельних виконань програми. Порівняйте ці значення для обох варіантів паралельного виконання.

4.4 Зміст звіту

1. Текст програми

2. Результати дослідження точності обчислень.

3. Результати досліджень обчислювальної складності для послідовного виконання, паралельного виконання за допомогою функцій Windows і паралельного виконання за допомогою Open MP.

4. Висновки

4.5 Контрольні запитання і завдання

1. Як включається режим використання Open MP?
2. Як виконуються директиви Open MP, якщо режим Open MP не вмикнуто?
3. Як визначається кількість ядер процесору?
4. Що таке прискорення для паралельного виконання програми?

5 ВИКОРИСТАННЯ ОБ'ЄКТІВ СІНХРОНІЗАЦІЇ ПРИ СТВОРЕННІХ ПРОГРАМ

5.1 Мета роботи

При створенні реальних багато поточних програм виникають проблеми синхронізації,