Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Уфимский университет науки и технологий»

Кафедра ВМиК

Отчёт по лабораторной работе №1

на тему: «ПАРАЛЛЕЛЬНОЕ ВЫЧИСЛЕНИЕ СУММЫ РЯДА СРЕДСТВАМИ АВТОМАТИЧЕСКОГО РАСПАРАЛЛЕЛИВАНИЯ И OPENMP»

по дисциплине: «Параллельные вычисления»

Выполнили:

Студент группы: ПРО-428Б Мохаммед А. А.

Аль-Шаибани Е. Т.

Вахитов Т. Ф.

Проверила: Шерыхалина Н.М.

Уфа – 2023

1. **Цель работы**

Научиться использовать основные директивы, функции и переменные окружения интерфейса OpenMP на примерах написания простейших параллельных программ для многоядерных вычислительных систем.

1. **Выполнение работы**
2. Выполнить программную реализацию на языке С / С++ последовательного алгоритма суммирования функционального ряда согласно варианту. Предусмотреть вывод на экран времени работы программы.
3. Произвести распараллеливание программы путём добавления директив OpenMP в текст последовательной программы.
4. Отладить написанные программы при небольшом числе членов ряда. Убедиться, что результаты последовательной и параллельной программ корректны и стабильны при многократных запусках. Убедиться, что количество порождаемых потоков равно числу ядер вычислительной системы.
5. Добавить опцию задания числа порождаемых потоков и запустить программу при различном числе потоков.
6. Подключить автоматическое распараллеливание к последовательной версии программы (ключ /QParallel). Убедиться, что количество порождаемых потоков равно числу ядер вычислительной системы.

A black text on a white background

Description automatically generated

A white sheet with black text

Description automatically generated

1. **Ход работы**

Задание 1:  
Дан ряд согласно варианту 2

A math equations and symbols

Description automatically generated with medium confidence

**Task 3: Sequential vs. Parallel Comparison**

The sequential and parallel algorithms are compared for different values of N. The results demonstrate the correctness of the parallelization and provide insights into the performance improvement achieved.

A screenshot of a graph

Description automatically generated

**Task 4: Varying Thread Counts**

The program is executed with varying thread counts to evaluate the scalability of the parallel algorithm. Results are presented, and the impact of different thread counts is analyzed.

A screenshot of a computer screen

Description automatically generated

**Task 6: Performance for Different N Values**

The performance of both sequential and parallel algorithms is evaluated for different values of N. The results provide an understanding of how the algorithm scales with the size of the problem.

A screenshot of a computer program

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| программа |  | Tp, c |  |
|  | N1 | N2 | N3 |
| Послед. | 26.739 | 53.97 | 113.769 |
| OpenMP | 4.771 | 9.476 | 20.735 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| программа |  | Tp, c |  |  | Tp, c |  |
|  | N1 | N2 | N3 | N1 | N2 |  |
| Послед. | 26.739 | 53.97 | 113.769 | 26.739 | 53.97 |  |
| OpenMP | 4.771 | 9.476 | 20.735 | 4.771 | 9.476 |  |

**Task 8: Load Balancing Strategies**

The program is executed with different load balancing strategies (static, dynamic, guided) and chunk sizes. The results are presented, and the impact on performance is discussed.

A screenshot of a computer

Description automatically generated

|  |  |  |  |
| --- | --- | --- | --- |
| Kind |  | TP , c |  |
|  | C1 | C2 | C3 |
| Static | 5.288 | 5.053 | 5.023 |
| Dynamic | 102.089 | 5.498 | 5.337 |
| guided | 5.238 | 5.405 | 5.261 |

1. **Выводы**

В ходе выполнения расчетно-графической работы мы научились

использовать основные директивы, функции и переменные окружения интерфейса OpenMP на примерах написания простейших параллельных программ для многоядерных вычислительных систем