НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра прикладної математики

Звіт

із лабораторної роботи №3

із дисципліни «Розподілені і хмарні обчислення»

|  |  |
| --- | --- |
| Виконав: | Керівник: |
| студент групи КМ-03 | Ліскін В. О. |
| Шаповалов Г. Г. |  |

Київ — 2023

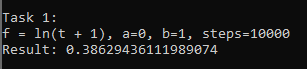
Мета роботи

Проінтегрувати методом редукії.

Проінтегрувати методом правих прямокутникв за різної кількості потоків та інтервалів розбиття.

Опис програми

Завдання 1 – порахуємо інтеграл в отопоточному режимі для визначення його значення. Кількість інтервалів розбиття =10000



Завдання 2 – змінюємо інтервали розбиття, кількість потоків та порівнюєм час

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Інтервали розбиття | Потоки | Кращий час виконання | Середній час виконання | Гірший час виконання |
| 10 | 1 | 1.8464 µs | 1.8943 µs | 1.9565 µs |
| 10 | 2 | 1.9286 µs | 2.1238 µs | 2.3294 µs |
| 10 | 4 | 2.4385 µs | 2.4856 µs | 2.5377 µs |
| 10 | 8 | 3.6720 µs | 3.7353 µs | 3.8050 µs |
|  | 1 | 60.605 µs | 63.460 µs | 67.404 µs |
|  | 2 | 33.189 µs | 34.054 µs | 35.143 µs |
|  | 4 | 15.294 µs | 15.617 µs | 15.996 µs |
|  | 8 | 23.131 µs | 23.453 µs | 23.770 µs |
|  | 1 | 41.794 ms | 45.173 ms | 48.580 ms |
|  | 2 | 26.001 ms | 26.126 ms | 26.262 ms |
|  | 4 | 6.8540 ms | 7.1791 ms | 7.5387 ms |
|  | 8 | 3.4796 ms | 3.5489 ms | 3.6233 ms |

Лістинг програми:

use std::{sync::Arc};

use lab\_3::integration::integral\_reduction;

use std::process::Command;

fn main() {

    testing();

    task\_1();

    task\_2();

}

fn task\_1() {

    println!("\n\n\nTask 1:");

    let f = |t: f64| (t + 1.0).ln();

    let a = 0.0;

    let b = 1.0;

    let steps = 10000;

    println!("f = ln(t + 1), a={}, b={}, steps={}", a, b, steps);

    let af = Arc::new(f);

    let res = integral\_reduction(&af, a, b, steps);

    println!("Result: {}", res);

}

fn task\_2() {

    println!("\n\n\nTask 2:");

    println!("Запускаємо cargo bench");

    match Command::new("cargo").args(&["bench"]).status() {

        Ok(status) => if status.success() {

            println!("cargo bench виконано успішно");

        } else {

            eprintln!("cargo bench завершилося з помилкою");

        },

        Err(e) => eprintln!("Помилка при запуску cargo bench: {}", e),

    }

}

fn testing() {

    println!("\nЗапускаємо cargo test");

    match Command::new("cargo").args(&["test"]).status() {

        Ok(status) => if status.success() {

            println!("cargo test виконано успішно");

        } else {

            eprintln!("cargo test завершилося з помилкою");

        },

        Err(e) => eprintln!("Помилка при запуску cargo test: {}", e),

    }

}