**Trabalho 1.1 - FORK**

Dupla: Rita de Cássia Lino Lopes - T03

Társila Samille Santos da Silveira - T01

**Passos para compilar:**

1. Abrir terminal
2. Digitar o seguinte comando:

g++ integral.cpp -o integral -pthread

**Passos para executar:**

1. Após compilar o programa
2. No terminal, digitar o seguinte comando:

./integral

1. Digitar valores para t (número de thread) e n (número de trapézios).

**Dados do Sistema Operacional:**

Distributor ID: Ubuntu

Description: Ubuntu 20.04.2 LTS

Release: 20.04

Codename: focal

**Versão do GCC:**

9.3.0

Abaixo há uma cópia do código completo:

#include <iostream>

#include <cstdlib>

#include <pthread.h>

#include <math.h>

using namespace std;

#define PI 3.14159265

double a = 0, b = 10; // f1

//double a=0, b=2\*PI; //f2

double h, sum = 0, integral;

int n, t, trapezio\_por\_thread;

double \* x, \* y;

struct thread\_data {

int thread\_id;

};

double f1(double x) {

return 5;

}

double f2(double x) {

return sin(2.0 \* x) + cos(5.0 \* x);

}

void \* fun(void \* threadarg) {

struct thread\_data \* my\_data;

my\_data = (struct thread\_data \* ) threadarg;

int inicio = my\_data -> thread\_id \* trapezio\_por\_thread;

int fim = (inicio + trapezio\_por\_thread) >= t ? t : (inicio + trapezio\_por\_thread);

for (int i = inicio; i < fim; i++) {

x[i] = a + i \* h;

y[i] = f1(x[i]);

if (i!=0 && i!= t) sum += h \* y[i];

}

pthread\_exit(NULL);

}

int main(int argc, char \* argv[]) {

cout << "Digite o numero de threads:";

cin >> n;

cout << "Digite o numero de trapezios:";

cin >> t;

int rc;

pthread\_t threads[n];

struct thread\_data td[n];

h = (b - a) / t;

trapezio\_por\_thread = ceil((double)t / n);

x = new double[t];

y = new double[t];

for (int i = 0; i < n; i++) {

td[i].thread\_id = i;

rc = pthread\_create( & threads[i], NULL, fun, (void \* ) & td[i]);

if (rc) {

cout << "Error," << rc << endl;

exit(-1);

}

}

for (int i = 0; i < n; i++) {

pthread\_join(threads[i], NULL);

}

integral = h / 2.0 \* (y[0] + y[t - 1]) + sum;

cout.precision(2);

cout<<scientific;

cout << "\nA integral é " << integral << endl;

pthread\_exit(NULL);

}

As figuras 1 e 2 mostram, respectivamente, o resultado da execução dos testes. O teste com número de threads igual a 6 e de trapézios a 120 e o segundo teste com 7 threads e 200 trapézios.

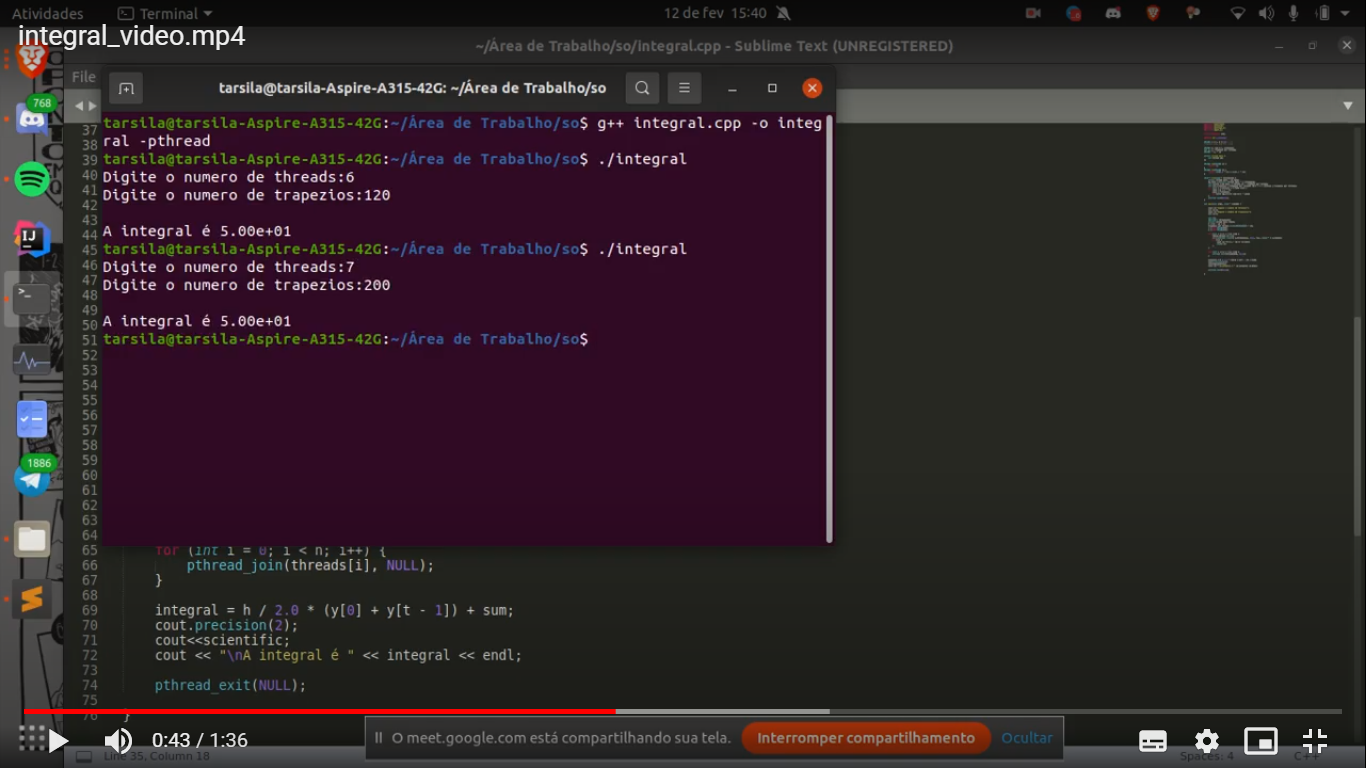
Figura 1: Resultado dos testes na função 1

Figura 2: Resultado dos testes na função 2