Saint Petersburg National Research University of Information Technologies, Mechanics and Optics (ITMO University)
Faculty of Informational Technologies and Programming

Report
about laboratory work №1
«Definite integral calculation»

Student
Sultan Zhumabaev J4133c

St. Petersburg
2020

1. GOAL OF LABORATORY WORK

Calculate the value of a definite integral with precision ε.

2. TASK DEFINITION

The Computational Problem:

1 Choose precision ε.

2 Calculate integral with different A and B values from the table.

3 Calculate execution time of serial program.

4 Write a parallel program with:

- atomic;
- critical sections;
- locks;
- reduction.

5 Count speedup with different thread number.

6 Fill the table (for each point of 4).

3. BRIEF THEORY

The *atomic* directive ensures that a specific memory location is updated atomically, rather than exposing it to the possibility of multiple, simultaneous writing threads.

The *critical* directive identifies a construct that restricts execution of the associated structured block to a single thread at a time.

OpenMP 3.0 specifies that *locks* are no longer owned by threads, but by tasks. Once a *lock* is acquired, the current task owns it, and the same task must release it before task completion. The change in *lock* ownership requires extra care when using *locks*.

Reduction performs a reduction on the scalar variables that appear in *variable-list*, with the operator *op*. The syntax of the *reduction* clause is as follows: *reduction*(*op* : variable-list). A *reduction* is typically specified for a statement with one of the following forms:

- $x = x\ op\ expr$;
- $x\ binop = expr$;
- $x = expr\ op\ x$ (except for subtruction);
- $x++$;
- $++x$.

4. ALGORITHM (METHOD) OF IMPLEMENTATION

The algorithm for numerical calculation of a certain integral with a given accuracy is shown in the figure 1.

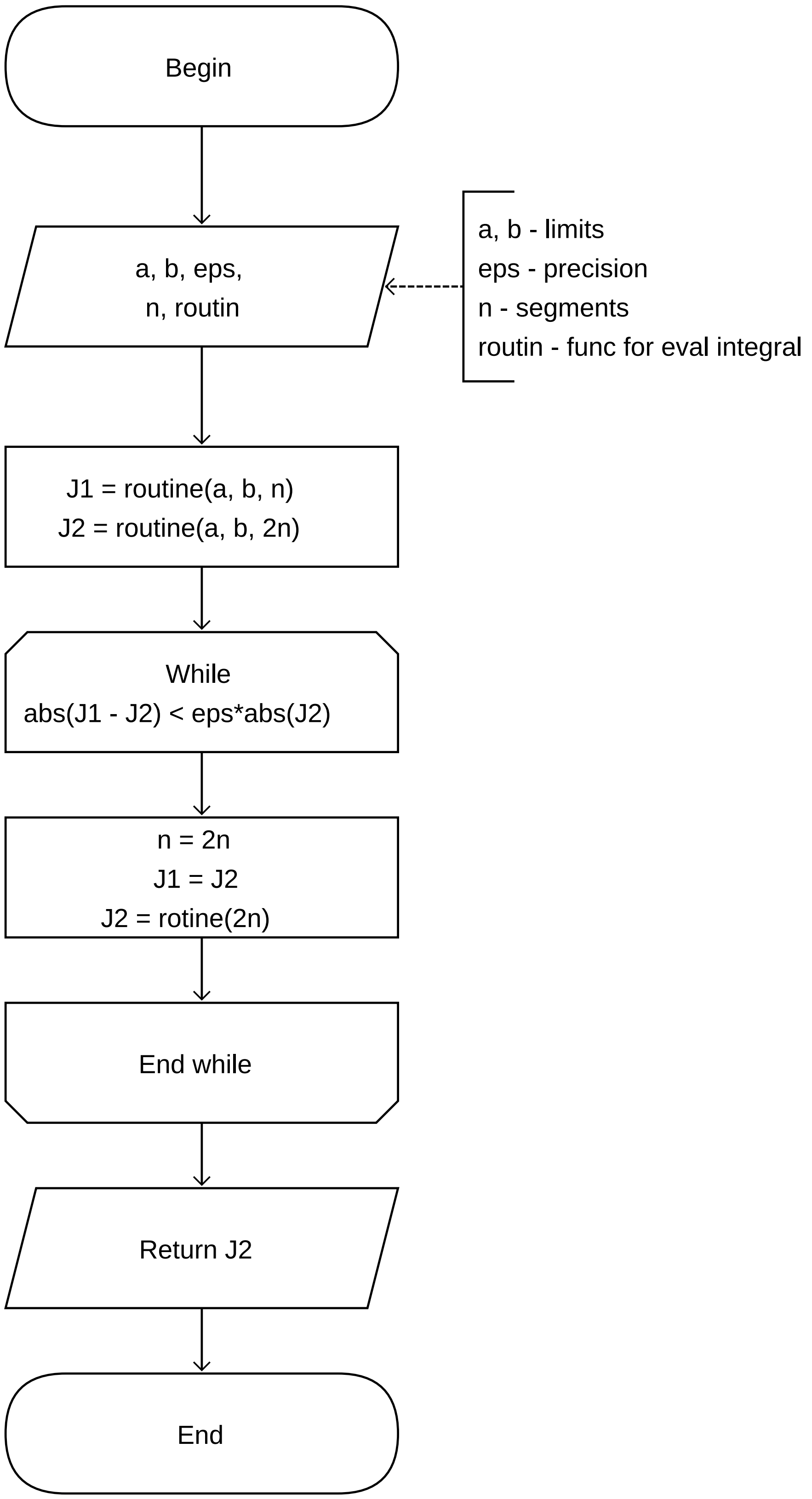

Figure 1. Diagram of the algorithm

5. RESULT AND EXPERIMENTS

The results of measurements for a computer with 4 physical and 8 logical cores and a specified accuracy of 1e−6 are shown below (Table 1).

Table 1. Average results of 200 measurements

			time, ms				
a	b	threads	serial	atomic	critical_section	locks	reduction
1e-5	1e-4	4	0.294	0.291	0.307	0.295	0.305
		8	0.315	0.309	0.303	0.305	0.310
		16	0.310	0.303	0.313	0.306	0.316
1e-4	1e-3	4	0.146	0.147	0.154	0.153	0.147
		8	0.147	0.158	0.155	0.153	0.159
		16	0.155	0.167	0.158	0.164	0.156
1e-3	1e-2	4	0.079	0.073	0.073	0.072	0.072
		8	0.076	0.073	0.074	0.075	0.076
		16	0.076	0.078	0.077	0.084	0.076
0.01	0.1	4	0.038	0.037	0.036	0.037	0.040
		8	0.037	0.037	0.037	0.040	0.037
		16	0.037	0.036	0.037	0.037	0.037
0.1	1	4	0.035	0.035	0.035	0.037	0.035
		8	0.035	0.035	0.037	0.035	0.036
		16	0.035	0.035	0.035	0.039	0.037
1	10	4	0.003	0.003	0.003	0.003	0.003
		8	0.003	0.003	0.003	0.004	0.003
		16	0.003	0.003	0.003	0.003	0.003
10	100	4	0.003	0.003	0.003	0.003	0.003
		8	0.003	0.003	0.003	0.003	0.003
		16	0.003	0.003	0.003	0.003	0.003

In theory, the best result should be given by the "reduction"operation on 8 threads, but due to the simplicity of the task and the CPU load of background tasks, the results were ambiguous.

Based on the description of other methods for providing exclusive access, threads should be blocked for addition operations when calculating the amount. Thus, this section is critical and the execution time should increase.

It is also worth noting that with 16 threads, pseudo-parallelism begins, when one thread takes resources from another.

6. CONCLUSION

You have to carefully select the appropriate blocking methods for each task. Depending on the computer, the optimal number of threads may vary.

7. APPENDIX

The source code is located here: `https://github.com/vanSultan/parallel_algorithm/tree/lab_01/lab_01`.