Programming Parallel Computers

Aalto 2023

IS9a: better algorithm ★★★

Please note that you can still submit, but as the course is already closed, your submissions will not be graded.

To get started with the development, **download the code templates**, unzip the file, edit is.cc, and run `./grading test` or `./grading benchmark` to try it out — see the **instructions** for more details!

Upload your solution as a file here...

Please upload here the file **is.cc** that contains your solution to task IS9a.

Choose File No file chosen

... or copy-paste your code here

Submit

Your submissions

Your submissions to IS9a will appear here; you can simply **reload** this page to see the latest updates.

What you will need to do in this task

Please read the **general instructions for this exercise** first. Here are the additional instructions specific to this task:

> Design a more efficient algorithm that (at least in typical cases) does not need to try out all possible locations of the rectangle. Implement the algorithm efficiently on the CPU.

What I will try to do with your code

I will first run all kinds of tests to see that your code works correctly. You can try it out locally by running `./grading test`, but please note that your code has to compile and work correctly not only on your own computer but also on our machines.

If all is fine, I will run the benchmarks. You can try it out on your own computer by running `./grading benchmark`, but of course the precise running time on your own computer might be different from the performance on our grading hardware.

Benchmarks

Name	Parameters
benchmarks/1 the input is a multicolor image with 100 × 100 pixels	nx = 100, ny = 100
benchmarks/2a the input is a structured black-and-white image with 199 × 199 pixels	nx = 199, ny = 199
benchmarks/2b the input is a structured black-and-white image with 200 × 200 pixels	nx = 200, ny = 200
benchmarks/2c the input is a structured black-and-white image with 201 × 201 pixels	nx = 201, ny = 201
benchmarks/3 the input is a multicolor image with 400 × 400 pixels	nx = 400, ny = 400
benchmarks/4 the input is a multicolor image with 1000 × 1000 pixels	nx = 1000, ny = 1000

Grading

In this task your submission will be graded using benchmarks/4: the input is a multicolor image with 1000 × 1000 pixels.

The point thresholds are as follows. If you submit your solution no later than on **Sunday, 04 June 2023, at 23:59:59 (Helsinki)**, your score will be:

Running time	Points
≤ 10.000 sec	1
≤ 5.000 sec	2
≤ 2.000 sec	3
≤ 1.000 sec	4
≤ 0.500 sec	5

If you submit your solution after the deadline, but before the course ends on **Sunday, 04 June 2023, at 23:59:59 (Helsinki)**, your score will be:

Running time	Points
≤ 5.000 sec	1
≤ 2.000 sec	2
≤ 0.500 sec	3