Operating Systems
Jacobs University Bremen
Dr. Jürgen Schönwälder

Module: CO-562
Date: 2023-01-08
Due: 2023-01-15

OS 2022 Problem Sheet #13

This sheet is only for students who failed to obtain the module achievement.

Problem 13.1: *multi-threaded factorization* (10 points)

Write a multi-threaded program called `factorize`, which factorizes natural numbers concurrently. The program reads the numbers to be factorized from the standard input and writes the result to the standard output. Multiple numbers can be factorized concurrently using the thread pool pattern. The `-t` command line argument controls how many worker threads are created, i.e., the size of the thread pool. A sample invocation is shown below:

```
$ seq 1 4747 123456 | cat -n | sed -e 's/^[[:blank:]]*//' > /tmp/input
$ ./factorize -t 6 < /tmp/input
1 1 = 1
3 9495 = 3 * 3 * 5 * 211
5 18989 = 17 * 1117
2 4748 = 2 * 2 * 1187
6 23736 = 2 * 2 * 2 * 3 * 23 * 43
7 28483 = 7 * 13 * 313
8 33230 = 2 * 5 * 3323
10 42724 = 2 * 2 * 11 * 971
11 47471 = 37 * 1283
12 52218 = 2 * 3 * 3 * 3 * 967
9 37977 = 3 * 12659
14 61712 = 2 * 2 * 2 * 2 * 7 * 19 * 29
4 14242 = 2 * 7121
13 56965 = 5 * 11393
17 75953 = 151 * 503
18 80700 = 2 * 2 * 3 * 5 * 5 * 269
15 66459 = 3 * 22153
20 90194 = 2 * 13 * 3469
21 94941 = 3 * 3 * 7 * 11 * 137
22 99688 = 2 * 2 * 2 * 17 * 733
16 71206 = 2 * 35603
24 109182 = 2 * 3 * 31 * 587
25 113929 = 59 * 1931
23 104435 = 5 * 20887
26 118676 = 2 * 2 * 29669
19 85447 = 85447
27 123423 = 3 * 41141
```

Note that every number is prefixed with a tag, which is used to match results to requests. The content of the tag is arbitrary but it is safe to assume that it does not contain white space or control characters. Detailed requirements:

a) The program has to factorize numbers in the unsigned 64-bit integer number space correctly.

b) The program must use the thread pool pattern where a fixed number of threads are created to perform the factorization.

c) The tags and numbers are read from the standard input by an input reading thread and placed into a queue where they are picked up by worker threads.

d) Factorization results are passed via another queue to an output writing thread.

e) The program must detect the end of the input and shutdown the worker threads.

f) The main thread waits until all worker threads, the reader thread, and the writer thread have terminated.

g) The program must follow the programming requirements stated on the course web page. In particular, it needs to handle runtime error situations in an appropriate manner.