

GEBZE TECHNICAL UNIVERSITY
COMPUTER ENGINEERING

SYSTEM PROGRAMMING LECTURE

HOMEWORK 5

REPORT 5

AHMET FURKAN KURBAN

1801042674

MAY 23

Design Decisions and Problem Solved Part:

Data was read from the file. If the number of data in the file is not equal to $2^n \times 2^n$ an error is returned. Threads created. timestamp written in seconds. Clock function was used for time differences. Each Thread calculated the AXB of its own column. Then, before going to the 2nd stage, each thread was expected to calculate AXB. For this, condition variable and mutex were used. Then the 2nd stage was passed and 2d Discrete Fourier Transform was calculated for the same column. Then the threads finished their work. The results were written to the file and exited as free. When I check with Valgrind, sometimes in sigint it leaks and sometimes it doesn't. I saw that the error was caused by the pthread functions in the description. I made all mallocs free. However, this did not improve.

Which requirements I achieved:

I correctly calculated the AXB and the 2D furier transform from the data read from the file. I have written the results in File.

Which requirements I failed:

I followed the requirements in the pdf one by one and I think that I am not missing anything.

RESULTS:

N=5

With 2 thread:

```
Seconds: 1653208723 The process has written the output file. The total time spent is 0.248456 seconds.
```

With 4 thread:

```
Seconds: 1653208831 The process has written the output file. The total time spent is 0.264500 seconds.
```

With 8 thread:

```
Seconds: 1653208851 The process has written the output file. The total time spent is 0.291434 seconds.
```

N=3

With 2 thread:

```
Seconds: 1653210547 The process has written the output file. The total time spent is 0.003775 seconds.
```

With 4 thread:

```
Seconds: 1653210638 The process has written the output file. The total time spent is 0.005305 seconds.
```

With 8 thread:

```
Seconds: 1653210671 The process has written the output file. The total time spent is 0.006665 seconds.
```

When I increased the number of threads, I saw that the completion of the process took longer. The reason for this may be that the process of creating threads is more than the process of dividing the process. I think that this extra thread will take less time in large inputs. It hasn't changed much in small inputs.

