Hi! This is the prompt for your Addepar take-home programming exercise. The rules are:

- You may use any tools you like, but please produce a solution we will be able to easily run on a single machine in a typical UNIX environment. Your solution should include:
    - All source code and any configs or other files necessary to build and run your program.
    - Full instructions for building and running your program. (Please include this in a README file.)
    - A quick analysis of the computational complexity and space complexity of your program. (Please include this in the same README file.)
- When you are ready to submit a solution, zip up your full solution and reply to this e-mail attaching the zip.
- You may use Google, Stack Overflow, etc. for questions about specific library calls, compiler errors, etc., but please do not search for solutions to the overarching problem we're asking you to solve.
- There is no strict deadline, but please try to send a timely response. We aren't going to agonize about the difference between a submission that arrives in 2.5 hours and one that arrives in 3, but if you submit 8 hours after receiving the prompt we're likely to wonder what took all that time.

For this exercise, we would like you to **write a program to merge a directory of sorted text files.**

Each file in the directory contains a number of lines of text. All of the non-empty lines of text within each text file should be sorted in lexicographic order. Your job is to write a program that takes in the name of the directory and an output location, looks at the files in the directory, and produces a text file in the output location that:

- Contains no empty lines of text
- Contains no duplicate lines
- Contains each unique non-empty line of text contained by any file in the directory, in lexicographic order

Your solution should also *verify* that the input criteria are true. If one of the input files is *not* sorted by line in lexicographic order, your solution should print an appropriate error and exit.

For example, for the command:

```
./merge input_dir output_file
```

...where `input_dir` contains the following files and contents:

```
a.txt:
```

```
      familiarity breeds contempt

      slow and steady wins the race
      the pen is mightier

      the truth will set you free

b.txt:
      no peace for the wicked
      no peace for the wicked
      to err is human

c.txt:
      no peace for the wicked


      the truth will set you free
```

The output file created should contain:

```
      familiarity breeds contempt
      no peace for the wicket
      slow and steady wins the race
      the pen is mightier
      the truth will set you free
      to err is human
```

Please think carefully about the performance characteristics of your solution. It should work efficiently even if the directory contains a very large number of files. It should work efficiently even if the files in the directory are very large.

We're looking forward to seeing your solution. The prompt should hopefully be enough to get started, but if you run into any serious questions please shoot us an e-mail. Thanks!