
Update: 06.12.2022

What to put in the project report:

How your code should be used. You need to specify the input arguments and their sequence
Methodology on how zip/unzip was implemented and parallelized.
Mention any blocks of code which were critical sections.
Mention any optimizations you did in the code to improve performance and reduce memory usage.

Two additional txt documents have to be submitted:

- Memory usage and error report which will be given by valgrind
- Multithreading error report given by valgrind --tool=helgrind

Evaluation metrics for your project will be:

- Methodology of how multithreading was performed.
- Any performance and memory optimizations which were done
- The handling of exceptions. E.g. what happens if less/more command line arguments are given. What happens if the number of threads is <=0…. etc.
- Heap usage. The lower the better.
- Stack should not be used extensively. You should not store large arrays in the stack. Such stack usage will be penalized.
- Memory leaks and errors will be penalized.
- Code which is not thread safe will be penalized.

Operating Systems Term Project

You are required to build a multi-threaded zip and un-zip application. Following is the interface:

zip <text file name> <number of threads> <output file name>
unzip <zipped file name> <number of threads> <output file name>

The zip/unzip will be done using huffman coding. For tutorial on this, please go over the following references: geeksforgeeks, wikipedia, programiz
The programiz reference has code as well. You may use this code as a reference.

The steps for the zip application are the following

1. Parse through the text file and determine the frequency of different characters
2. Determine the huffman encoding using the characters and their corresponding frequencies
3. Using the encoding, compress the text file/

4. Save compressed file and the file containing encoding information. The file containing encoding information should be named <output_file_name>_encoding.bin
5. Steps 1-3 need to be multi-threaded.

The steps for the un-zip application are the following

1. Read the *_encoding.bin file to determine the binary encoding of different characters.
2. Decode the characters in the zipped file.
3. Write the output in the <output_file name> file.
4. Step 2 needs to be multithreaded

We will use an instrumentation tool to measure the memory footprint and performance of your code. The specifics of the tool and the criteria will be shared later. Your code will be evaluated on:

- Correctness
- Memory footprint
- Performance