COSC 3360 (16770) - COSC 6310 (11209) - Operating Systems

Dashboard / My courses / COSC3360SP202502 / PROGRAMMING ASSIGNMENTS / Programming Assignment 1

Description Submission view

Available from: Monday, 13 January 2025, 12:00 AM
Due date: Sunday, 16 February 2025, 11:59 PM
Requested files: main.cpp (Download)
Type of work: Individual work

This programming assignment closes at 11:58:59 PM on 02/16/2025.

Similarity Threshold: 95%

Problem:

For this assignment, you will create a multithreaded program using the Encoding Method for Sparse Binary Data presented by Weiwei Duan et al. (2017) (A New Spatial Encoding Method for Sparse Binary Data) to draw a 2D image with printable ASCII characters.

Given a multiline input representing the image to decode using the Encoding Method for Sparse Binary Data with the following information per line:

1. Image size: two positive integer values greater than zero with the size of the image (length and width). The integer values are separated by a single white space.
2. Output characters: a comma-separated line with the symbols to print. Each symbol will be followed by the range(s) in the x-axis where it will be printed (a symbol can have multiple ranges). The symbol (printable ASCII characters) and its range(s) (at least two positive integer values) are separated by a single white space. The information about each character is separated by a comma.
3. The values for the headPos array (review the paper above). This line has multiple positive integer values (including zero) separated by a single white space.

4. The values for the dataPos array (review the paper above). This line has multiple positive integer values (including zero) separated by a single white space.

Input example:

```
26 7
U 0 10,H 15 25
0 4 8 12 25 29 33
0 10 15 25 0 10 15 25 0 10 15 25 0 10 15 16 17 18 19 20 21 22 23 24 25 0 10 15 25 1 9 15 25 2 3 4 5 6 7 8 15 25
```

The expected output is:

```
U         U    H         H
U         U    H         H
U         U    H         H
U         U    HHHHHHHHHHH
U         U    H         H
 U       U     H         H
  UUUUUUU      H         H
```

Process:

Your solution must execute the following steps:

- Read the input lines from STDIN.
- Create n POSIX threads (where n is the number of lines in the image (width)). Each child thread executes the following tasks:
  - Receives the line number to decode, the image size, the headPos and dataPos arrays, the information about the symbols to print from the main thread, and the memory location to store the decoding process results.
  - Decodes the assigned line using the Encoding Method for Sparse Binary Data.
  - Stores the decoded line on a memory location accessible by the main thread.
- Print the decoded image into STDOUT.

Notes:

- You can safely assume that the input will always be in the proper format.
- You must use the output statement format based on the example above.
- You can define additional functions if needed.
- You must take full advantage of multithreading.
- You must present code that is readable and has comments explaining the logic of your solution. A 10% penalty will be applied to submissions not following this guideline.
- You cannot use global variables. A 100% penalty will be applied to submissions using global variables.
- Not using multiple POSIX threads to implement your solution will translate into a penalty of 100%.
- A penalty of 100% will be applied to solutions that do not compile.
- A penalty of 100% will be applied to solutions hardcoding the output.

Assignment rubric:

- Correct output: 10 points per test case (20 points total).
- Correct implementation of the Encoding Method for Sparse Binary Data: 20 points.
- Taking full advantage of multithreading (no pthread_join or sleep in the same loop as pthread_create): 30 points.
- Creating the correct number of threads: 20 points.
- Having clean and commented code: 10 points.

Penalties:

1. Presenting a solution that does not compile: -100 points.
2. Not using POSIX threads: -100 points.
3. Hardcoding the output: -100 points.
4. Using global variables: -100 points.

Requested files

main.cpp

```
// Write your program here
```