Java Concurrent Programming

CS331: Programming Languages Laboratory

Lab-1 Q4 Report

Task:

Sorting using multithreading

Author:

Gautam Sharma, 210101042

Instructor:

Prof. Sukumar Nandi, Dept. of CSE, IITG

Contents

1 Introduction

This lab was the first of the **11 labs** in the course **CS331: Programming Languages Laboratory**. This was an introductory lab, with **Multithreading**, and **Synchronization** in Java as the main topics being touched upon.

2 Task

To sort an array of integers using multiple threads.

3 Input

The user has to enter **two things**:

- The `size` of the array.
  The `size` should be less than x?
- The `contents` of the array.
  The `contents` of the array can be integers ranging from `-2147483648` to `2147483647`.

```
Enter the size of the array:
6
Enter the contents of the array:
23476 253 123 56 87 1
```

Figure 1: Sample Input

4 Approach

Merge Sort was used to sort the array. For sorting each `left` and `right` parts of an array, **two threads** were spawned.

```
Initial array: [23476, 253, 123, 56, 87, 1]
Spawned left and right threads for: [23476, 253, 123, 56, 87, 1]
Spawned left and right threads for: [23476, 253, 123]
Spawned left and right threads for: [56, 87, 1]
Spawned left and right threads for: [253, 123]
Spawned left and right threads for: [87, 1]
```

Figure 2: Sample processing

5 Output

The sorted array is given as the output to the user.

```
Sorted array: [1, 56, 87, 123, 253, 23476]
```

Figure 3: Sample output

6 Technical Documentation

The task was implemented in the class `Q4` of the file `Q4.java`.

6.1 Public Variables

There are no public variables!

6.2 Private Variables

6.2.1 ANSI_RESET

Type: `constant`, `static`, `String`
Usage: Resetting output color to white

6.2.2 ANSI_RED

Type: `constant`, `static`, `String`
Usage: Changing output color to red

6.2.3 ANSI_GREEN

Type: `constant`, `static`, `String`
Usage: Changing output color to green

6.2.4 ANSI_YELLOW

Type: `constant`, `static`, `String`
Usage: Changing output color to yellow

6.2.5 ANSI_CYAN

Type: `constant`, `static`, `String`
Usage: Changing output color to cyan

6.3 Public Classes

There are no public classes!

6.4 Private Classes

6.4.1 sorter

Type: Implements `Runnable`
Attributes: `int[] array` - the array to be sorted
Usage: Spawns threads to sort an array

6.5 Public Methods

6.5.1 main

Type: `static`, `void`
Arguments: None
Usage: Taking the input and solving the task by calling the private methods.

6.6 Private Methods

6.6.1 sort

Type: `void`
Arguments: `int[] array`
Usage: Divide the array into two parts if the array size is greater than one, and spawn two threads for the left and right parts for sorting them, and after that call the `merge` method to merge the parts.

6.6.2 merge

Type: `void`
Arguments: `int[] left_part`, `int[] right_part`, `int[] array`
Usage: Merges the `left_part` and `right_part` of the array in non-decreasing order.

7 Program flow

The program undergoes the following steps:

1. Enters the `main` method, takes input and calls the `sort` method.
2. Enters the `sort` method, which spawns two threads to sort the left and right parts of the array using `sorter` class, and then later merges the sorted parts using `merge` method.
3. Final output is printed.

8 Usage

Refer to the README.