CS401 Lab 5: Recursion and Algorithm Implementation

Overview

- This lab is to be completed individually.
- Focus: Understanding and implementing recursive and iterative algorithms.
- Objectives:
    - Implement the *Jacobsthal* sequence using recursive and iterative approaches.
    - Create a recursive method to find the minimum value in an array.

Part 1: Jacobsthal Sequence

Background

The *Jacobsthal* sequence is an integer sequence that follows this recurrence relation:
- J(0) = 0
- J(1) = 1
- J(n) = J(n-1) + 2J(n-2) for n > 1

The first few numbers in the sequence: 0, 1, 1, 3, 5, 11, 21, 43, 85, 171, 341, 683, 1365, 2731, ...

Requirements

1. Implement two Java methods in a class named *Jacobsthal*:
   - *long jacobsthalRecursive(int n)*
   - *long jacobsthalIterative(int n)*
2. Both methods should return the nth *Jacobsthal* number.
3. In the *main* method:
   - Accept an integer input n from the command line.
   - Calculate and print the first n *Jacobsthal* numbers using both methods.
   - Measure and display the execution time for each method.

Expected Output

```
$ java Jacobsthal 10
Recursive version: 0, 1, 1, 3, 5, 11, 21, 43, 85, 171
Time taken to execute recursive version: XX.XX milliseconds

Iterative version: 0, 1, 1, 3, 5, 11, 21, 43, 85, 171
Time taken to execute iterative version: XX.XX milliseconds
```

Part 2: Recursive Minimum Finder

Requirements

1. Implement a recursive method in a class named *Minimum* with the following signature:

```
  public static int minimum(int A[], int size)
```

   ```

2. The method should return the smallest value in the first *size* elements of the array *A*.
   ```

3. Use the following framework to complete your implementation:

```
public class Minimum {
    public static int minimum(int A[], int size) {
        // Implement your recursive algorithm here
    }

    public static void main(String args[]) {
        int A[] = {10, -20, 1, 2, 0, 5, 100};
        int s = minimum(A, A.length);
        System.out.println(s);
    }
}
```

Expected Output

```
$ java Minimum
-20
```

Submission Requirements

1. Source Code:
   - Jacobsthal.java
   - Minimum.java
2. Compiled Bytecode:
   - Jacobsthal.class
   - Minimum.class
3. Output:
   - PDF file containing program outputs for both parts
4. Executable JAR file
5. README file

Important Notes

- Ensure your recursive implementations have proper base cases to avoid infinite recursion.
- For the Jacobsthal sequence, consider potential integer overflow for large inputs.
- Test your programs with various inputs, including edge cases.