**Dúvidas que foram surgindo durante a implementação - Lógica**

Houve dúvida na implementação da lógica, uma vez que foi considerado o uso de recursividade para a resolução do problema. Inicialmente, essa abordagem parecia intuitiva, mas revelou-se inadequada devido ao aumento exponencial do número de chamadas recursivas sem memorização, com um grau de complexidade (O(2ⁿ)) e risco de *stack overflow*. Assim, optei pela solução iterativa, que mantém os valores calculados em memória e possui complexidade linear (O(n)), garantindo eficiência, simplicidade e reutilização de cálculos.

Para combater as chamadas em simultâneo que poderiam causar inconsistência nos valores calculados ou até mesmo na expansão do array, optei por usar o modificador *synchronized* na implementação do método *calculateLabseq*. Assim faz com que seja seguro o acesso concorrente, uma vez que apenas pode aceder ao método uma thread de cada vez. Garantindo assim que operações críticas são realizadas de forma ordenada e sem interferência.

Surgiram ainda dúvidas acerca de qual estrutura de dados seria a melhor para armazenar a sequencia já calculada até ao momento: Arrays, ArrayList ou HashMap. Comecei por fazer a implementação com arrays, mas cada vez que era chamado um índice que ainda não constava na sequência, era necessário o redimensionamento do array, isto é, copiar todos os valores para um novo array com o tamanho do índice pedido (n). Seria uma abordagem a priorizar a eficiência em memória, mas quando o tamanho do array não cresce muito. Como no caso, o índice da sequência cresce frequentemente e linearmente, a escolha optada foi ArrayList, combinado a simplicidade do código sem tanta sobrecarga de memória comos seria exigida pelo HashMap.