

Problem Set 9

Note: Try to make the best use of appropriate C++ features.

You are provided with a working implementation of a SkipList that stores integer keys. Your task is to extend this implementation to support generic types using C++ templates and then use it to develop a simple LogSystem that manages log entries with integer timestamps and string messages.

Objective:

- Modify the given **SkipList** and **Node** classes to support **generic types using C++ templates**
- It should support **SkipList<Key, Value>**, for example **SkipList<Int, String>**
- Using your SkipList<int, std::string> implementation, create a LogSystem class to efficiently store and manage logs.
- The LogSystem should at least support the following operations:
 - void insert(int timestamp, const std::string& message);
Inserts a new log entry into the system. The entry is indexed by the given timestamp and stores the associated message. If the timestamp already exists, you may choose to overwrite or ignore it (design-dependent).
 - std::string search(int timestamp) const;
Searches for a log entry with the specified timestamp.
 - Returns the corresponding message if found.
 - Returns "NOT FOUND" if the timestamp does not exist in the log system.
 - bool remove(int timestamp);
Removes the log entry with the given timestamp from the system.
 - Returns true if the entry existed and was successfully deleted.
 - Returns false if the timestamp was not found.
 - void display() const;
Displays all log entries in ascending order of timestamps.
 - Each entry should include the timestamp and its corresponding message.
- Use the provided main function to test your implementation.