Data Structures and Introduction to Algorithms - 20407

Maman 13

Yehonatan Simian
206584021

Overview

The project contains of four main files:

- `mergeable_heap.h` the declaration of the interface.
- `sorted.h` an implementation using sorted linked-lists.
- `unsorted.h` an implementation using unsorted linked-lists.
- `game.hpp` a user interface as requested in the instructions.

The files are well documented, thus I will not repeat the documentation here. In addition to the documentation in the code, the `docs.pdf` and `docs-compact.pdf` files are provided with a Doxygen generated documentation.

However, I have decided to provide a complexity table that summarizes the entire project:

Operation	UnsortedLinkedHeap	SortedLinkedHeap	LazyBinomialHeap
MAKE-HEAP	O(1)	O(1)	O(1)
INSERT	O(1)	O(n)	O(1)
MINIMUM	O(1)	O(1)	O(1)
EXTRACT-MIN	O(n)	O(1)	O(log n) amortized
UNION	O(1)	O(n+m)	O(1)

Table 1: Complexity Summary

Note that I have added a lazy binomial heap data structure to the table. The reason is that I have accidentally implemented this data structure before reading further instructions in the forum, and I decided to keep this implementation (available via `lazy.h`) because it's both elegant and efficient.

Usage

In order to run the program, all one must do is

```cpp
#include "game.hpp"
int main()
{
  Game game{};
  game.run();
}
```

The code above is available via the provided `main.cpp`, and can be compiled using the following instruction: `g++ -std=c++23 -Wall -Wextra -Werror -Wpedantic -o main main.cpp`. I recommend using gcc 13.1.0 or later versions.

A compiled version of main is available via the provided `main` binary.