

# Raport Proiect 3

---

## Comparing linked lists and skip lists

### Descrierea codului

---

Proiectul a fost implementat in limbajul C++ folosind programare orientata pe obiecte. Am implementat doua clase sablon, `SkipList` si `SortedLinkedList`, in functie de tipurile de date folosite pentru chei si pentru valori. In plus, am adaugat restictia ca tipul de date folosit pentru cheie sa fie un tip de date comparabil.

Structurile de date au implementate trei functionalitati comune: inserarea unui nod in lista (pastrand ordinea sortata a cheilor), stergerea unui nod din lista dupa valoarea cheii si afisarea tuturor elementelor din liste. Motivul pentru care nu am implementat si o functie de cautare a unui element este ca aceste cautari sunt efectuate implicit la inserari si stingeri de noduri.

### Teste si experimente

---

Pentru testare, am folosit ca tip de date pentru cheie un string de 10 litere. Am generat o lista de 10000 de chei unice, asociate cu un numar (numarul asociat nu este foarte relevant deoarece nu afecteaza sub niciun fel performanta programului). Lista generata avea cheile amestecate. Am incarcat, pe rand, toate datele mai intai in *linked list*, apoi in *skip list*, urmand sa efectuez mai apoi si stergerea datelor folosind toate cheile anterioare. Performanta am masurat-o folosind utilitare din biblioteca `chrono`.

Rezultate in modul Debug:

```
linkedList construction time: 330 ms
skipList construction time: 126 ms
linkedList total removal time: 250 ms
skipList total removal time: 15 ms
```

Rezultate in modul Release:

```
linkedList construction time: 105 ms
skipList construction time: 46 ms
linkedList total removal time: 105 ms
skipList total removal time: 5 ms
```