Implementacja tablicy haszującej

Arkadiusz Janus

1. **Złożoność obliczeniowa poszczególnych metod.**

    - bool insert(const int &x);
    - **O(N),** ponieważ dodajemy element na koniec listy musimy przejść ją całą.
      Operacja dominująca: porównanie
    - bool remove(const int &x);
    - **O(N),** ponieważ obliczamy hash w czasie O(1), i w najgorszym przypadku znajdziemy element na końcu listy w O(N).
      Operacja dominująca: porównianie
    - bool contains(const int &x);
    - **O(N),** ponieważ obliczamy hash w czasie O(1), i w najgorszym przypadku znajdziemy element na końcu listy w O(N).
      Operacja dominująca: porównianie
    - int hash(const int &x);
    - **O(1),** ponieważ jest to prosta operacja artymetyczna.
      Operacja dominująca: modulo
    - void makeEmpty();
    - **O(M*N),** ponieważ musimy przejrzeć wszystkie klucze w czasie O(N), i wszystkie elementy o tych kluczach których też może być dowolna ilość M.
      Operacja dominująca: porównianie
    - void print();
    - **O(M*N),** ponieważ musimy przejrzeć wszystkie klucze w czasie O(N), i wszystkie elementy o tych kluczach których też może być dowolna ilość M.
      Operacja dominująca: porównianie