**Київський національний університет імені Тараса Шевченка Факультет комп’ютерних наук та кібернетики**

**Кафедра інтелектуальних програмних систем**

**Алгоритми та складність**

**Завдання № 8**

**Варіант № 3**

**Звіт**

**Виконав:**

студент групи К-29

Грищенко Юрій Анатолійович

**Київ-2020**

**Завдання**

Реалізація піраміди Фібоначі.

**Предметна область**: відділ кадрів (варіант 3). Об’єкти: відділення фірми, працівники. Маємо множину відділень, у кожному відділенні зберігається множина працівників.

**Теорія**

* Слабша структура, ніж у біноміальної піраміди.
* Амортизований час операцій, що не використовують видалення, дорівнює O(1).
* Можуть виявитися корисними в алгоритмах, де частка операцій EXTRACT\_MIN та DELETE відносно мала (ряд алгоритмів на графах: наприклад, пошук мінімального кістякового дерева, найкоротший шлях з однієї вершини, алгоритми, що використовують DECREASE\_KEY для кожного ребра в майже повних графах).
* Складність реалізації, порівняно з бінарними (та k-арними) пірамідами, вища.
* Значення констант у формулах часу виконаннятакож високі.

**Структура даних**

Невпорядкований набір дерев з коренем, кожне з яких є незростаючою пірамідою. Кожен вузол x містить вказівник на батька p[x] і вказівник на одного з синів child[x]. Дочірні вузли вершини об’єднані в двозв’язний циклічний список (child list). (Операції видалення елемента та об’єднання двох таких списків займають константний час.) Кожен дочірній вузол y має вказівники на лівого та правого своїх братів: left[y], right[y]. Порядок братських вузлів довільний.

Для піраміди Фібоначчі H позначимо

**t(H)** – кількість дерев у списку коренів H,

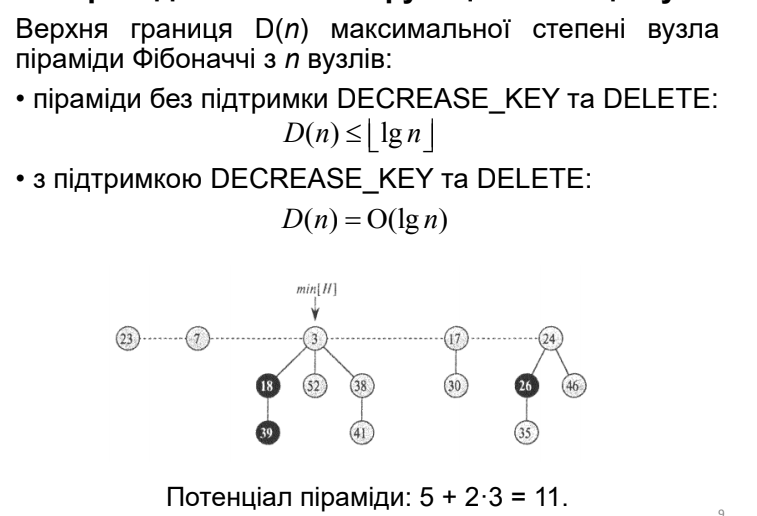
**m(H)** – кількість вузлів з мітками в піраміді H.

Тоді потенціал піраміди визначається як

Ф(H) = t(H) + 2m(H).

• Потенціал множини пірамід – сума потенціалів пірамід-складників.

Вважаємо, що одиниці потенціалу достатньо для покриття вартості довільної операції часу О(1). На початку роботи піраміда порожня, тобто початковий потенціал дорівнює 0. Тоді в подальшому потенціал завжди буде залишатися невід’ємним. Верхня границя загальної амортизованої вартості є верхньою границею загальної фактичної вартості послідовності операцій.



Якщо підтримуються лише операції MAKE\_HEAP, INSERT, MINIMUM, EXTRACT\_MIN та UNION, кожна піраміда Фібоначчі буде набором невпорядкованих біноміальних дерев (unordered binomial tree):

невпорядковане біноміальне дерево U0 складається з єдиного вузла;

невпорядковане біноміальне дерево Uk складається з двох невпорядкованих біноміальних дерев Uk-1, причому корінь одного з них є довільним дочірнім вузлом іншого.

**Особливість пірамід Фібоначчі:** об’єднання дерев у піраміді буде відбуватися лише під час операції EXTRACT\_MIN.

**Властивості невпорядкованих біноміальних дерев:**

Невпорядковане біноміальне дерево Uk

1. має 2k вузлів;

2. має висоту k;

3. має Сki вузлів на глибині i = 0,1...k;

4\* має корінь степеня k, а степінь інших вузлів буде менша; при цьому синами кореня є корені піддерев U0, U1, ..., Uk-1 в деякому порядку.

• Для піраміди Фібоначчі з n вузлами максимальна степінь вузла

****

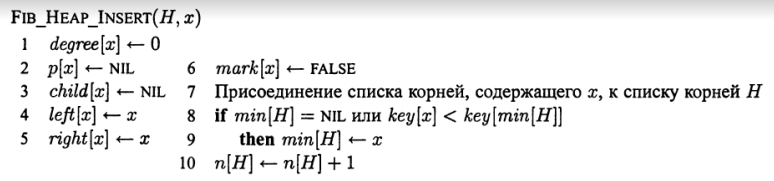
**Алгоритми**

**Створення піраміди Фібоначчі:**

Повертає нову порожню піраміду Фібоначчі H з n[H]=0

та min[H]=NIL. Час О(1).

**Вставка вузла:**



**Література**

* Лекція 6 з курсу “Алгоритми та складність” Шкільняк О.С. 2019-2020

Процедура не намагається об’єднати дерева в піраміді. Послідовне виконання FIB\_HEAP\_INSERT k разів призведе до додавання до списку коренів k дерев з одного вузла. Додавання вузла відбувається за **постійний час О(1).**

**Пошук мінімального вузла:**

На нього вказує min[H], тому пошук відбувається за час О(1).

**Об’єднання двох пірамід:**

Списки коренів пірамід Н1та Н2 просто об’єднуються і шукається новий мінімальний вузол.