Київський національний університет імені Тараса Шевченка

Факультет комп`ютерних наук та кібернетики

Кафедра інформаційних систем

Лабораторна робота №3

“D-арні піраміди”

Виконала студентка 2го курсу

Групи К-29

Шарова Ірина Валеріївна

2018

**Завдання**: d-арні піраміди схожі на бінарні, лише їх вузли, відмінні від листя, мають не по 2, а по d дочірніх елементів. Представте d-арну піраміду у вигляді масиву (якою буде її висота для n елементів?). Розробіть ефективні реалізації процедур Extract\_Max, Insert та Increase\_Key, призначених для роботи з d-арною незростаючою пірамідою. Проаналізуйте час роботи цих процедур і виразіть їх в термінах n та d.

• Висота d-арної піраміди для n елементів дорівнює

**Модулі програми:**

* int Parent (int i)

Визначає індекс батька і-го вузла

* int Child(int parent, int k)

Визначає індекс k-го сина у вузлі, батьком якого є parent

* void Balancing(vector<int> &Piramid, int i)

**Час роботи**: О(d\*)

Підтримує властивість незростання піраміди;

На вхід приймає масив Pyramid та індекс вузла і;

Вважається, що дерева з коренями Child(i, k) (k =1-d) –

незростаючі піраміди, але елемент Pyramid[i] може порушувати

цю властивість. Значення Pyramid[i] просувається вниз, поки відповідне дерево з

цим коренем не стане незростаючою пірамідою.

**Час роботи**: О(d\*)

На вхід приймає масив Pyramid;

Повертає і вилучає максимальне значення в піраміді;

За побудовою піраміди максимальне значення знаходиться у корені, воно вилучається, на його місце ставиться останній елемент масива і викликається процедура Balancing(Piramid, 0)

* void Increase\_Key(vector<int> &Piramid, int i, int key)

**Час роботи**: О()

На вхід приймає масив Pyramid, позицію і в піраміді та ключ key;

Змінює ключ елемента з індексом і на новий ключ key(за умови що значення нового ключа не менше за старе);

Після заміни ключа йде перевірка, чи не порушилась властивість незростання. В такому випадку відбувається обмін з батьківським елементом, поки властивість не відновиться.

* void Insert(vector<int> &Piramid, int key)

**Час роботи**: О()

На вхід приймає масив Pyramid та ключ key;

Вставляє в піраміду новий вузол зі значенням key;

**Тестові приклади:**

* d = 4

Piramid = {11, 30, 29, 28, 26, 25, 19, 16, 15, 13}

Збільшення ключа: і = 5, key = 40

Вставка елемента: key = 13

Графічне зображення роботи Balancing(Piramid, 0)

* d = 3 Piramid = {72, 160, 82, 144, 56,20, 30, 80, 74, 22, 21, 32, 126, 26}

Збільшення ключа: і = 6, key = 42

Вставка елемента: key = 31

**Література:**

* Лекція №5
* http://study.sfu-kras.ru/DATA/docs/ProgramTheory/recurs/s\_sort.htm