

## Лабораторная работа 5. Бинарная куча.

### Постановка задачи

Реализовать структуру данных «бинарная куча» (*min-heap*) на основе массива. Реализовать функции:

```
struct minheap *minheap_insert(struct minheap *heap,
                              int key, char *value);
void *build_minheap(int *arr, int n)
struct minheap *minheap_min(struct minheap *heap);
struct minheap *minheap_union(struct minheap *heap1,
                              struct minheap *heap2);
struct minheap *minheap_extractmin(struct minheap *heap);
struct minheap *minheap_decrease_key(struct minheap *heap,
                                     int key, int newkey);
struct minheap *minheap_delete(struct minheap *heap, int key);
```

Реализация функций бинарной кучи соответствует реализации в лекции.

### Экспериментальное исследование

- В качестве ключа использовать целое число типа *int*.
- Для добавления генерировать псевдослучайные числа в диапазоне [10 000, 1 000 000].
- При заполнении таблицы 1 необходимо изменить общее время добавления указанного количества элементов в структуру данных.
- Заполнение таблицы 2 предполагает, что сначала все элементы в указанном количестве будут добавлены в бинарную кучу.
- Для заполнения таблицы 2 необходимо извлечь элемент с минимальным приоритетом сразу после добавления всех элементов в структуру данных (*extractmin*), необходимо уменьшить ключ сразу после добавления всех элементов в структуру данных и извлечения одного элемента (*decreasekey*).

Таблица 1. Добавление элементов в структуру данных

#	Кол-во элементов	Время выполнения функции <i>minheap_insert</i> , с	Время выполнения функции <i>build_minheap</i> , с
1	50000		
2	100000		
3	150000		
.	...		
8	400000		

Таблица 2. Извлечение и уменьшение приоритета элемента в структуре данных

#	Кол-во элементов	Время выполнения функции <i>minheap_extractmin</i> , с	Время выполнения функции <i>minheap_decreasekey</i> , с
1	50000		
2	100000		
3	150000		
.	...		
8	400000		

---

## **Контрольные вопросы**

- Что такое очередь с приоритетом?
- Варианты реализации очереди с приоритетом.
- Объяснение механизма работы функций бинарной кучи.
- Вычислительная сложность операций бинарной кучи.
- Операция слияния двух куч.
- Как добавить элементы в бинарную кучу?
- Область применения бинарной кучи.