

## Bài tập Cấu trúc dữ liệu và giải thuật

**Bài 1.** Xét thuật toán tính giá trị của  $f(x,n)$  thể hiện trong hàm  $F(x,n)$  sau đây:

```
int F(int x, int n)
{
    if (n == 0) return 1;
    else if (n % 2 == 0) return F(x,n/2)*F(x,n/2);
    else return F(x,n/2)*F(x,n/2)*x;
}
```

Gọi  $T(n)$  là thời gian tính của thuật toán nói trên. Giả thuyết là các phép toán số học được thực hiện với thời gian bị chặn là hằng số.

- Xác định công thức đệ quy cho  $T(n)$ .
- Giải công thức đệ quy để đưa ra đánh giá của  $T(n)$  trong tình huống tồi nhất.

**Bài 2.** Đối với mỗi một trong các kiểu cấu trúc dữ liệu sau đây: Danh sách nối đơn, danh sách nối kép, hàng đợi dùng mảng. Hãy vẽ cấu trúc dữ liệu có được sau khi lần lượt bổ sung các phần tử của dãy các khóa: 4,2,6,7,6,5

### Bài 3.

- Biểu diễn cách sử dụng ngăn xếp để chuyển biểu thức dạng trung tố về dạng hậu tố:  
tổ:  $a - b * c ^ d - f$
- Hãy trình diễn cách tính giá trị của biểu thức hậu tố sau sử dụng ngăn xếp:  
 $1\ 2\ +\ 3\ 1\ +\ *\ 1\ 1\ +\ 1\ -\ /\$

**Bài 4.** Cho cây nhị phân ở hình bên. Hãy đưa ra thứ tự các đỉnh xác định bởi duyệt cây theo thứ tự trước, giữa, sau.



**Bài 5.** Cho mảng  $A=(0,2,4,3,8,9,6,5,7)$  biểu diễn 1 Min-heap.

- Vẽ cây nhị phân tương ứng với Min-heap đã cho.
- Trình bày các thao tác cần thực hiện trên cây để bổ sung thêm  $key=1$  vào min-heap nói trên để thu được 1 min-heap mới.

**Bài 6.** Struct TreeNode {

```
float key;  
struct TreeNode * LeftPtr;  
struct TreeNode * RightPtr;  
};  
typedef struct TreeNode BSTree;
```

a. Hãy viết hàm C sử dụng cấu trúc dữ liệu trên để thực hiện các thao tác sau đây với cây nhị phân.

- Tạo một nút mới.

```
BSTree *makeTreeNode(float value);
```

- Bổ sung một nút mới vào cây nhị phân tìm kiếm.

```
BSTree *insert(BSTree * nodePtr, float item);
```

b) Vẽ cây nhị phân tìm kiếm đối với tập các khóa  $S = (3, 2, 5, 4, 7, 6, 1)$  thu được nhờ thực hiện bổ sung lần lượt các khóa theo thứ tự đã cho vào cây nhị phân. Khởi tạo ban đầu là rỗng