

Bài tập tuần 8

Hãy cài đặt lớp BstSet biểu diễn một tập động các số nguyên bằng cây tìm kiếm nhị phân với các yêu cầu sau:

a) Cấu trúc một đỉnh của cây là Node:

```
struct Node{
    int key;           // khóa - là số nguyên
    Node * left;       // địa chỉ con trái
    Node * right;      // địa chỉ con phải
    Node * parent;     // địa chỉ cha
};
```

b) Quy ước dùng lá giả - tức các đỉnh lá không lưu khóa (đã bàn trong slides bài giảng) để việc cài các hàm cập nhật cây dễ dàng hơn.

c) Lớp BstSet có thành phần dữ liệu duy nhất là root (lưu địa chỉ của gốc) và các hàm public như sau:

1	BstSet()	Hàm kiến tạo không tham số: tạo tập rỗng
2	BstSet(char * filename)	Hàm tạo tập mới với dữ liệu là dãy số lưu trong tệp có tên filename
3	~BstSet()	Hàm hủy tập: giải phóng bộ nhớ động đã cấp phát cho tất cả các đỉnh của cây.
4	bool find(int x) const	Tìm xem x có trong tập không?
5	void erase(int x)	Nếu x có trong tập thì xóa x. Cây sau phép xóa vẫn là cây tìm kiếm nhị phân.
6	void insert(int x)	Nếu x không có trong tập thì chèn thêm x. Cây sau phép chèn vẫn là cây tìm kiếm nhị phân.
7	int min() const	Tìm khóa nhỏ nhất của tập.
8	int max() const	Tìm khóa lớn nhất của tập.
9	int predecessor(int x) const	Tìm khóa liền trước x.
10	int successor(int x) const	Tìm khóa liền sau x.
11	void printInorder() const	In cây theo thứ tự trong (inorder)

d) Viết hàm main để test tất cả các hàm của BstSet.