

## BÀI 1

Giả sử ta cần tính toán trên các số có dạng  $a + b\sqrt{7}$  với  $a, b$  là các số nguyên. Các thao tác trên kiểu số này bao gồm nhập, xuất, phép cộng, phép trừ, phép đổi dấu, phép nhân.

- Bạn hãy khai báo lớp đối tượng và các thao tác cho kiểu số này.
- Hãy cài đặt mã cho các thao tác kể trên.

## BÀI 2

Trình bày và cho ví dụ mã nguồn về *hàm hủy ảo* (*virtual destructor*) trong ngôn ngữ lập trình C++. Khi lập trình với ngôn ngữ lập trình C++, nếu không dùng hàm hủy ảo thì có nguy cơ gì? Bạn hãy cho ví dụ cụ thể có minh họa bằng mã nguồn.

## BÀI 3

Một *cấu trúc dữ liệu hướng đối tượng* được cài đặt bằng mã nguồn C++ như trong bảng sau. Để làm bài này, bạn cần đọc kỹ mã nguồn trước khi thực hiện các câu theo yêu cầu của đề bài.

<pre>#include &lt;iostream&gt; using namespace std; class absList { // Bắt đầu của lớp absList protected:     int dataId; public:     absList(int pId = 0) {         dataId = pId;     }     virtual ~absList() {}     int getData() {         return dataId;     }     virtual absList* addFirst(int pId) = 0;     virtual absList* getSubItem() = 0;     virtual void showAll(ostream&amp;) = 0;     virtual int countAll() {         return 0; // tạm thời 0 phần tử     } }; // Kết thúc lớp absList</pre>	<pre>class simpleList : public absList { public:     simpleList(int pId) : absList(pId) {     }     virtual absList* addFirst(int pId) {         dataId = pId;         return this;     }     virtual absList* getSubItem() {         return NULL; // Không có thành phần con     }     virtual void showAll(ostream&amp; outDev) {         outDev &lt;&lt; dataId &lt;&lt; " ";     }     virtual int countAll() {         return 1; // Chỉ có đúng một phần tử     } };</pre>
<pre>// Lớp linearList class linearList : public absList {     absList* subLst; public:     linearList(int pId) : absList(pId) {         subLst = NULL;     }     virtual ~linearList() {         if (subLst != NULL)             delete subLst;     } };</pre>	<pre>virtual absList* addFirst(int pId) {     linearList *Lst = new linearList(pId);     Lst-&gt;subLst = this;     return Lst; } virtual absList* getSubItem() {     return subLst; } virtual void showAll(ostream&amp; outDev) {     // Phương thức này chưa viết xong... } }; // Kết thúc linearList</pre>

```
1 void main() {  
2     simpleList *sLst = new simpleList(-13); absList *lnkLst = new linearList(37);  
3     for (int i = 1; i <= 8; i++) {  
4         lnkLst = lnkLst->addFirst(i * i - 7 * i);  
5     }  
6     delete lnkLst;  
7 }
```

- a) Có thể tạo ra đối tượng của lớp *absList* được hay không? Lớp *absList* được gọi là lớp gì?
- b) Dòng lệnh đánh nhãn số 6 (nghĩa là lệnh *delete lnkLst*) làm cho hàm hủy của lớp *linearList* chạy bao nhiêu lần? Hãy giải thích.
- c) Hiện tại lớp *linearList* sử dụng cài đặt mặc định của phương thức *countAll()* của lớp *absList* (trả về giá trị 0). Điều này không phù hợp vì phương thức này cần đếm số lượng các phần tử của cấu trúc dữ liệu *linearList*. Bạn hãy cài đặt lại mã nguồn của phương thức *countAll()*.
- d) Phương thức *showAll(ostream& outDev)* của lớp *linearList* chưa cài đặt xong. Bạn hãy cài đặt mã nguồn của phương thức này.
- e) Cấu trúc dữ liệu được cài đặt bởi lớp *linearList* thực chất là cấu trúc dữ liệu gì? Bạn có nhận xét gì thêm?