Midterm Exam

One and only question: A Vector is a collection of objects, each with an associated index. Objects may be inserted into a Vector at a specified index. Objects may be retrieved from the Vector by supplying an index. Insertion at an index containing an object overwrites the existing object.

Write a Vector class that stores and retrieves objects of the following class:

```cpp
class Object {
friend ostream & operator<<(ostream & out, Object & object);
    int number; bool null;
public:
    Object () { null = true; }
    Object (int n) { number = n; null = false; }
};
ostream & operator<<(ostream & out, Object & object) {
    if (object.null) out << "(empty)"; else out << object.number;
    return out;
}
```

The Vector class should have put (for insertion) and get (for retrieval) methods that are intended to be used as follows:

```cpp
Vector vec;
vec.put(10, new Object(23));
vec.put(3, new Object(13));
cout << *vec.get(3) << " " << *vec.get(88) << "\n";
```

which outputs 13 followed by (empty).

The Vector class objects should be linked lists of Cell objects:

```cpp
class Cell {
friend class Vector;
    Object *object; Cell *next; int index;
    Cell (Object *obj, Cell *nxt, int idx) {
        object = obj; next = nxt; index = idx;
    }
};
```

where each Cell object stores an Object object, points to a next Cell object, and stores the index of the Object object. All put invocations on indices without objects place the new Cell at the head of the list; if a Cell already exists at the specified index it is moved to the head of the list on a put.