```
#include <iostream>
#include<fstream>
#include<string>
#include <sstream>
using namespace std;

class ListNode {
	public:
		int pro;
		ListNode* next=NULL;
		char myChar;

		ListNode() {
			//The default constructor for not initializing any field
			this->next=NULL;

		}

		ListNode(char myChar,int pro) {
			this->myChar = myChar;
			this->pro = pro;
			this->next = NULL;
		}

};

class LinkedList {
	private:
		ListNode* head;
		ListNode* findSpot(ListNode *&newNode) {
			ListNode* current = head;
			while (current->next != NULL) {
				if ((newNode->pro) < (current->next->pro)) {
					break;
				}
				current = current->next;
			}
			return current;

		}

	public:
		LinkedList() {//constructor for creating a dummy head
			head = new ListNode();
		}
```

```
void listInsert(ListNode*& newNode) {
	ListNode* spot = findSpot(newNode);
	newNode->next = spot->next;
	spot->next = newNode;

}

string printList() { //the print function
	string result = "listHead->";
	ListNode* current = head->next;
	if (current != NULL) {
		char myChar = current->myChar;
		result =result+"('Dummy',0,"+"'"+myChar+"'"+")->";
	}
	else {
		result =result+ "('Dummy','0','NULL')->NULL";
		//cout << result << endl;
		//write to the file
		return result;
	}
	while (current->next != NULL) {

		char newChar = current->myChar;
		int pro = current->pro;
		ostringstream s;
		s  << pro;
		string newPro(s.str());

		//cout<<pro<<endl;
		//cout<<charPro<<endl;
		char nextChar = current->next->myChar;
		result = result + "("+"'"+newChar+"'," +newPro +",'"+nextChar+")->";
		current = current->next;
	}
	char newChar = current->myChar;
	int pro = current->pro;
	ostringstream s;
	s  << pro;
	string newPro(s.str());
	result = result + "(" + "'" + newChar + "'," +newPro  + ",'" + "'NULL'" + ")->";
	result = result + "NULL";
	cout << result << endl;
	//write to file
	return result;

}

~LinkedList() { //destructor for memory release
```

```
                while (head != NULL) {
                        ListNode* old = head;
                        head = head->next;
                        delete old;
                }

        }


};



int main(int argc, char** argv)
{
        //cout<<argv[1]<<"dd"<<endl;
        //cout<<argv[2]<<"dd"<<endl;
        ofstream output1;
        ifstream input;
        input.open(argv[1]);
        output1.open(argv[2]);

        LinkedList list;
    while (!input.eof())
        {

                char chr;
                int pro;
                input >> chr;
                input >> pro;
                if (input.eof()) break;
                ListNode* node = new ListNode(chr, pro);
                list.listInsert(node);

        }
        string result=list.printList();
        output1<<result;
        output1.close();
        input.close();

        //list.printList();
   return 0;
}
```