Practice 4: Drawing Lists

Given the class **Node** and the class **AnyList** as shown below, trace the code of the member function **createList** of the **AnyList** class, and draw the final singly-linked list produced, including **all** pointers.

Re-arrange the list so that it is in order (first node pointing to second node, second node pointing to third node, etc.). Your drawing must **show where the pointers are** pointing to.

```
class Node
{
public:
	Node() : data('X'), next(nullptr) {}
	Node(char theData, Node *newNext)
		: data(theData), next(newNext){}
	Node* getNext() const { return next; }
	char getData( ) const { return data; }
	void setData(char theData)
		{ data = theData; }
	void setNext(Node *newNext)
		{ next = newNext; }
	~Node(){}
private:
    char data;
    Node *next; //pointer pointing to next node
};
```

```
class AnyList
{
public:
	... //other functions
	void createList();
private:
	Node *first; //pointer pointing to
		     //first node

	// Note that there is NO count.
};
```

```
void AnyList::createList_1a()
{
	Node *p1 = nullptr;
	Node *p2 = nullptr;
	p1 = new Node('A', nullptr);
	p2 = new Node;
	p2->setData('B');
	Node *p3 = p1;
	p1->setNext(p2);
	p3->setData('C');
	first = p1;
	p3 = new Node('D', nullptr);
	p3->setNext(p1->getNext());
	p1->setNext(p3);
	Node *temp = nullptr;
	temp = new Node;
	temp->setData('E');
	temp->setNext(p1);
	p2 = nullptr;
	p3 = p1;
	first = temp;
	first = new Node('Y', first);
}
```