**Министерство высшего образования и науки Российской Федерации**

**ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ (ПНИПУ)**

**Электротехнический факультет**

**Кафедра Информационные технологии и автоматизированные системы**

**Отчёт по теме**

**«Лабораторная работа № 11»**

**«Информационные динамические структуры»**

**«Двунаправленный список»**

**Выполнил:**

**Студент 1 курса**

**Группы ИВТ-22-2б**

**Корючкин Савелий**

**Научный руководитель:**

**Доцент кафедры ИТАС**

**Полякова Ольга Андреевна**

**Пермь 2022**

**Постановка задачи**

Написать программу, в которой создаются динамические структуры и выполнить их обработку.

Записи в линейном списке содержат ключевое поле типа \*string. Сформировать однонаправленный список. Удалить из него К элементов с указанными номерами. Добавить К элементов с указанными номерами.

**Алгоритм решения**

1. Определить структуру узла списка struct Node. В узле содержатся ключевое слово типа char и указатель на следующий узел.
2. Определить структуру двунаправленного списка с указателями на начало и конец списка.
3. Реализовать функцию void addNode() для добавления элементов в конец списка.
4. Реализовать функцию void print\_list() для вывода списка на экран.
5. Реализовать функцию Node\* del\_Node(), в которой создается новый узел, указатель устанавливается на текущий элемент, текущий элемент удаляется и ссылка на текущий элемент перемещается на следующий.
6. Реализовать функцию void add\_Node для добавления элементов справа и слева от ключевого слова. Функция при этом учитывает граничные случаи, такие как вставка перед головным элементом и вставка в конец списка.
7. Реализовать функцию Node\* delete\_list(), которая полностью удаляет все узлы в связанном списке, начиная с узла beg, используя функцию Node\* del\_Node() для удаления узлов.
8. Реализовать функцию void printINfile(), которая записывает информацию из связанного списка, начиная с узла beg, в текстовый файл с указанным путем. Если файл не удалось открыть, выводится сообщение об ошибке.
9. Реализовать функцию Node\* readFROMfile(), которая считывает информацию из текстового файла с указанным путем и создает связанный список, содержащий прочитанные значения. Если файл не удалось открыть, выводится сообщение об ошибке.
10. В функции int main() произвести тестирование всех функций по постановке задачи.

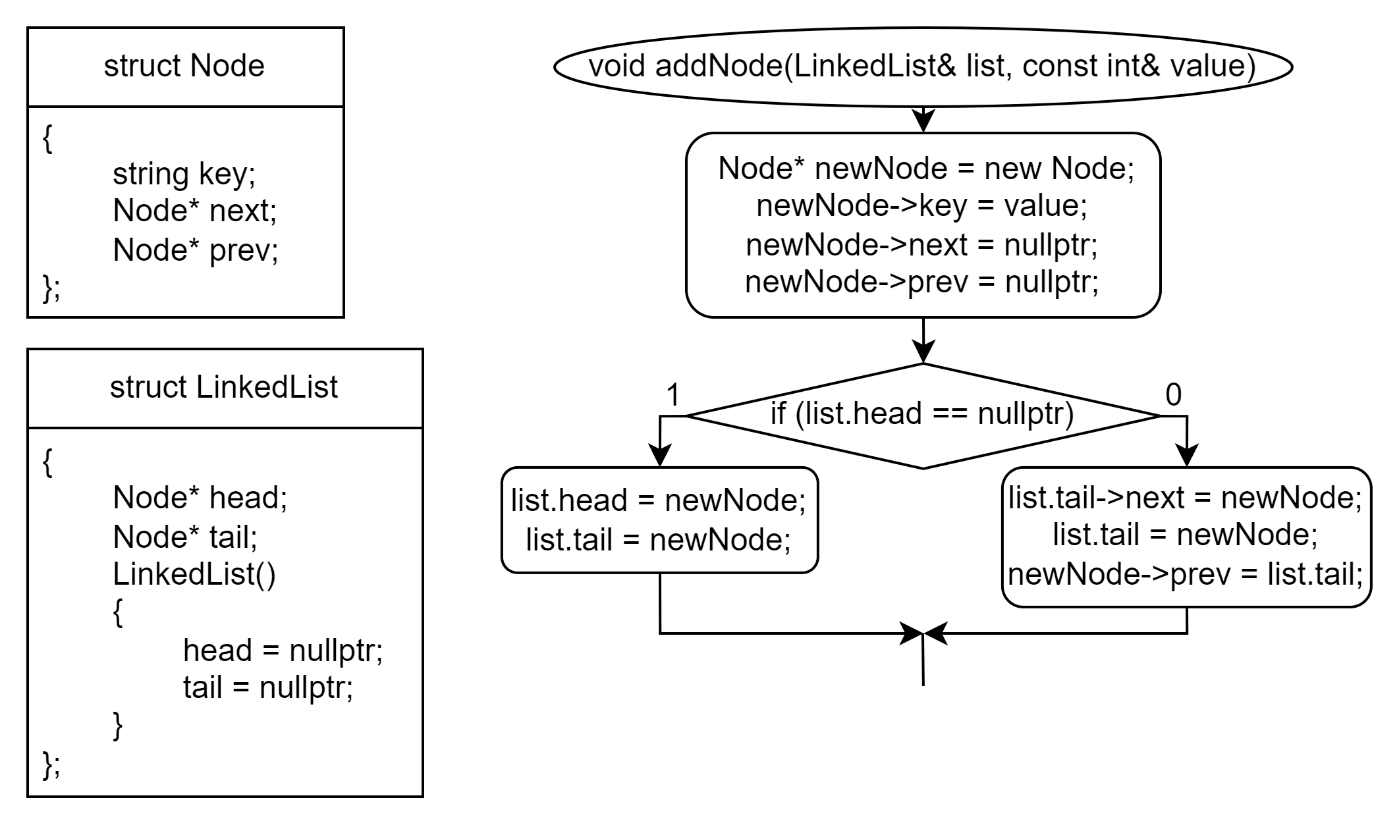


Рис. 1 – Структура узла struct Node, структура списка struct LinkedList, функция void addNode()

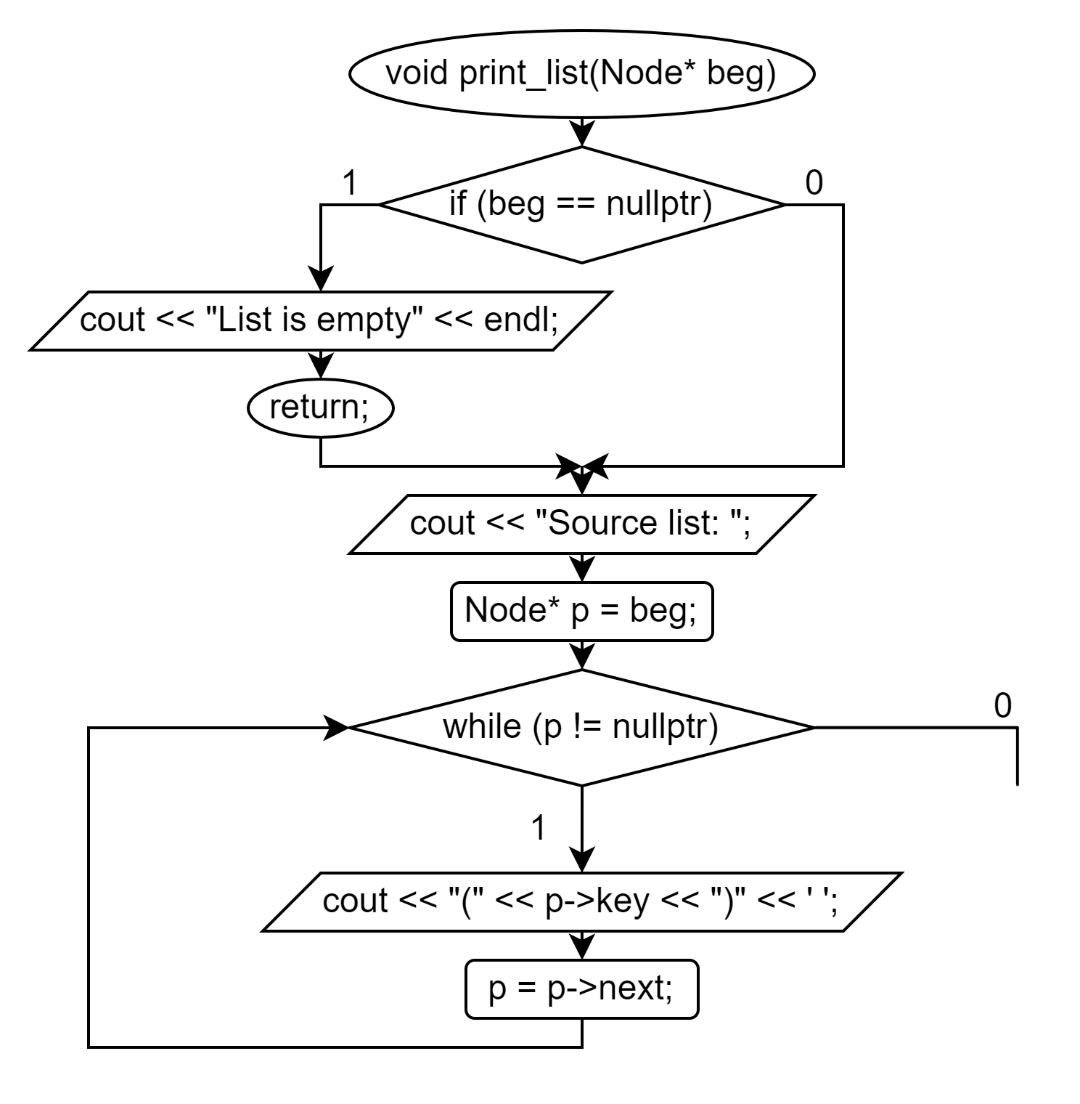


Рис. 2 – Функция void print\_list()

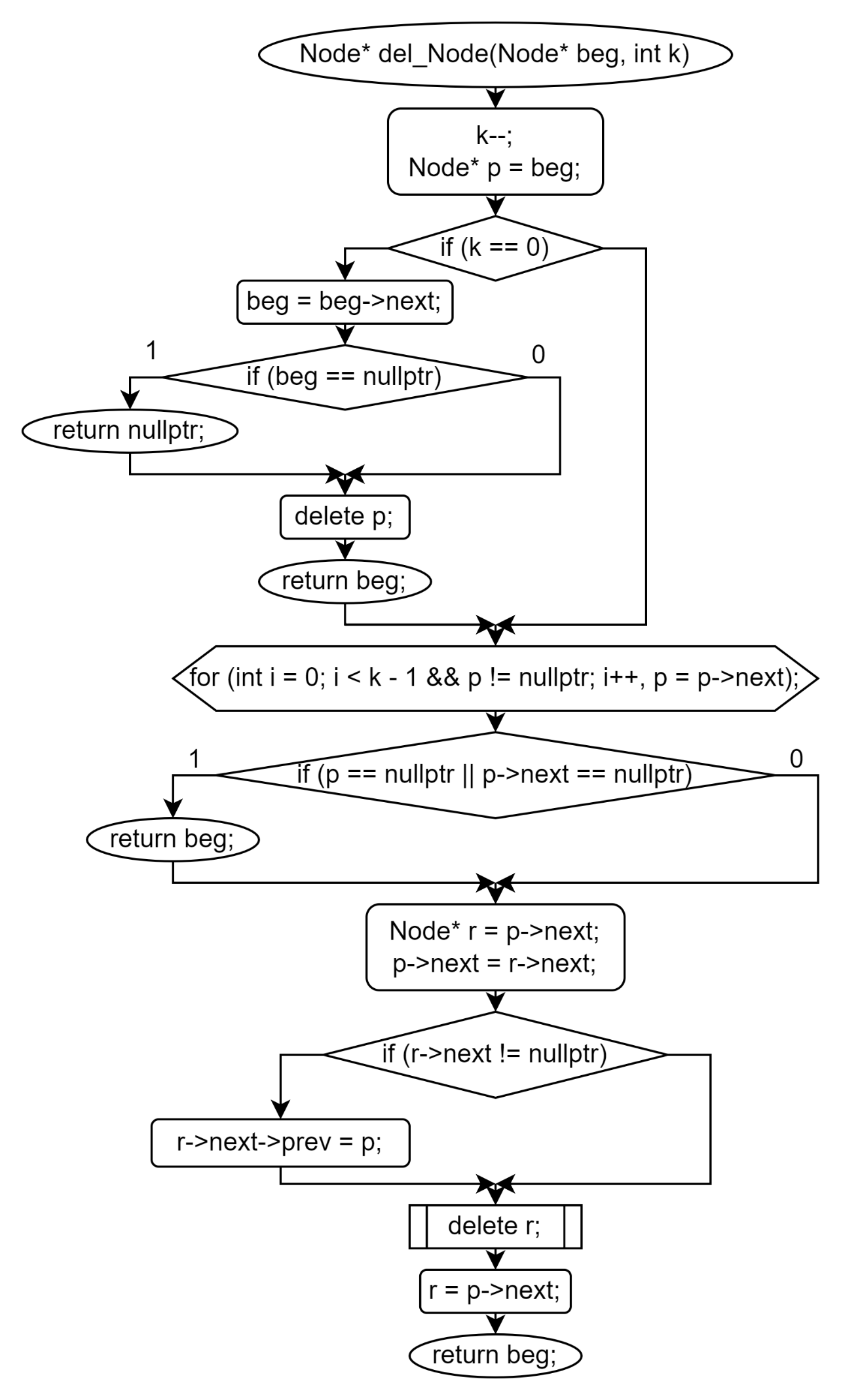


Рис. 3 – Функция Node\* del\_Node()

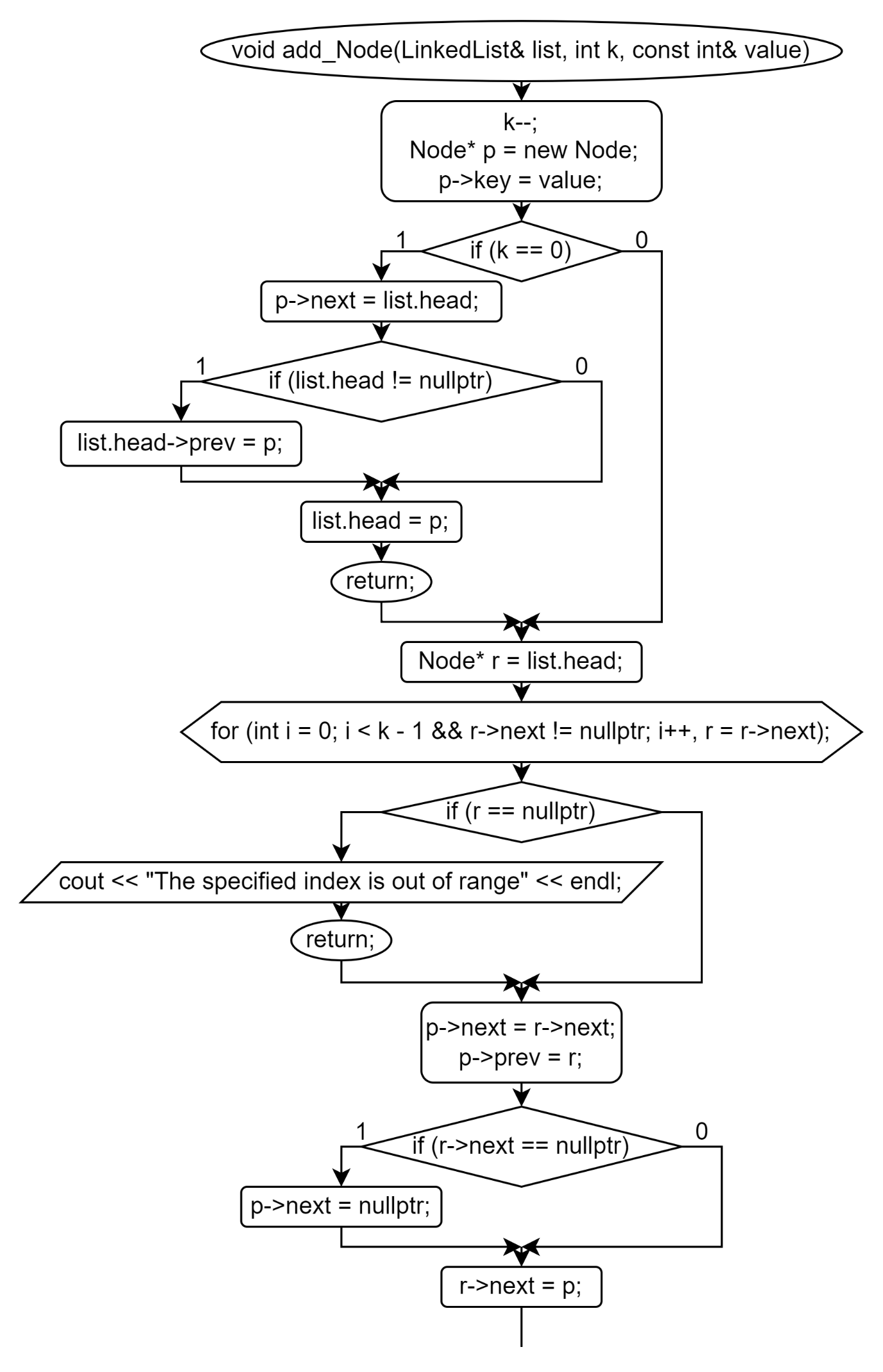


Рис. 4 – Функция void add\_Node()

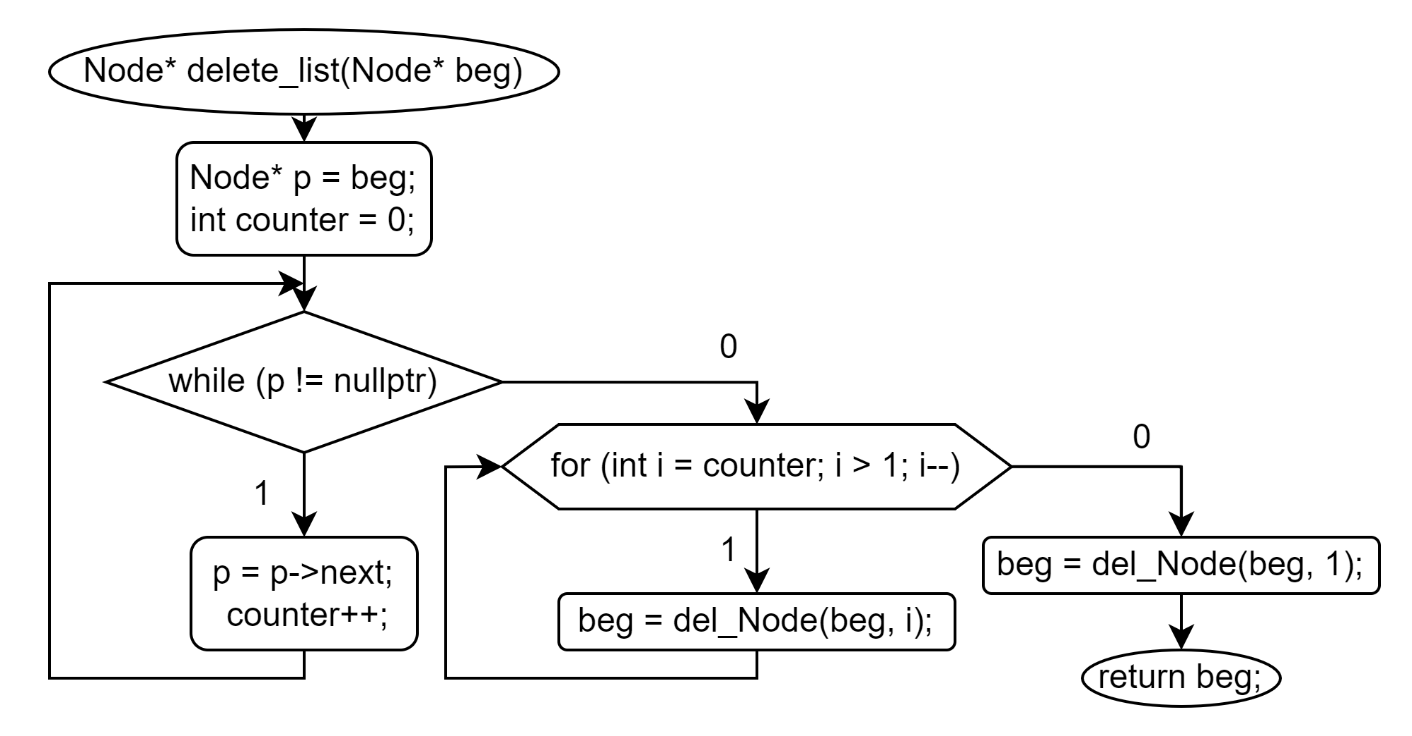


Рис. 5 – Функция Node\* delete\_list()

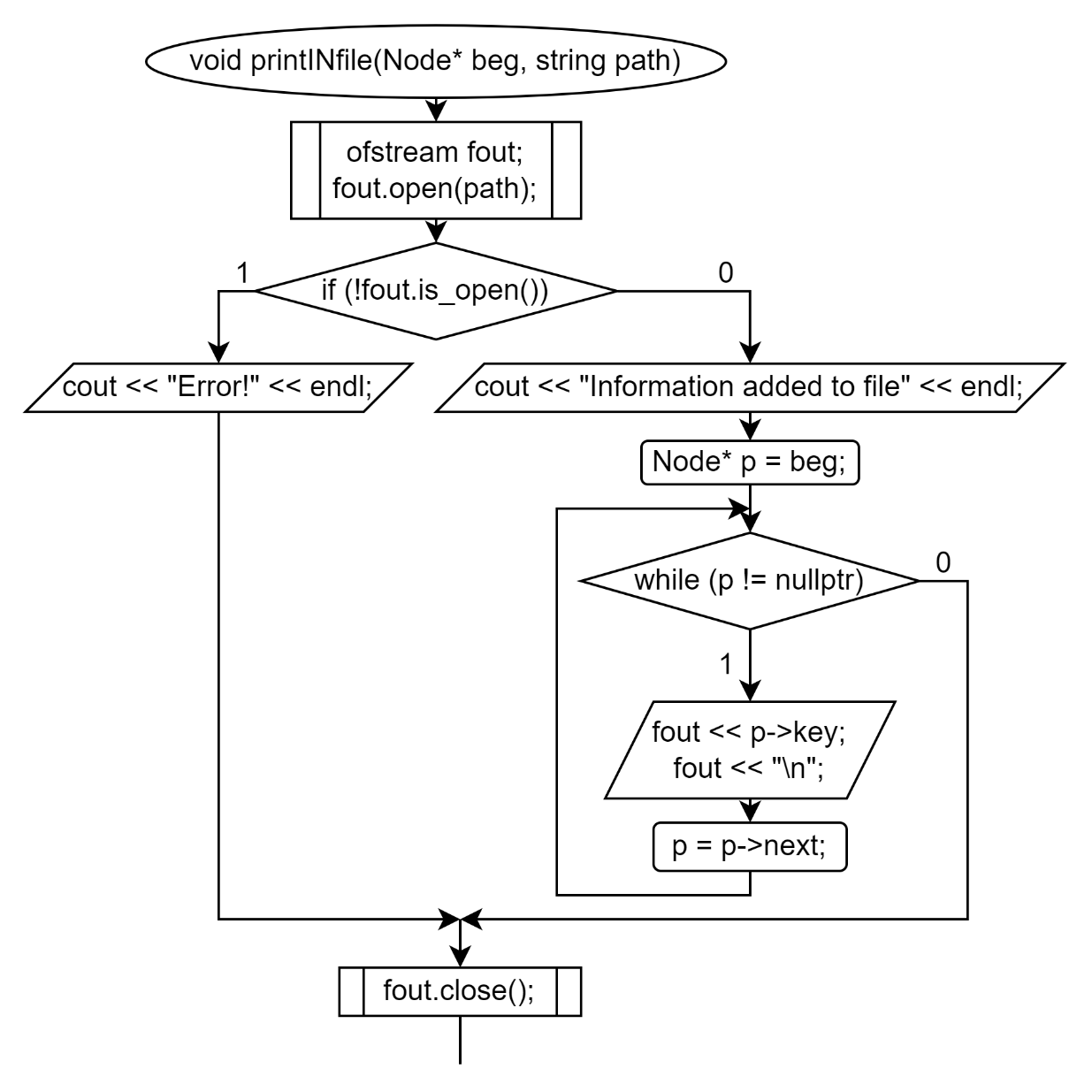


Рис. 6 – Функция void printINfile()

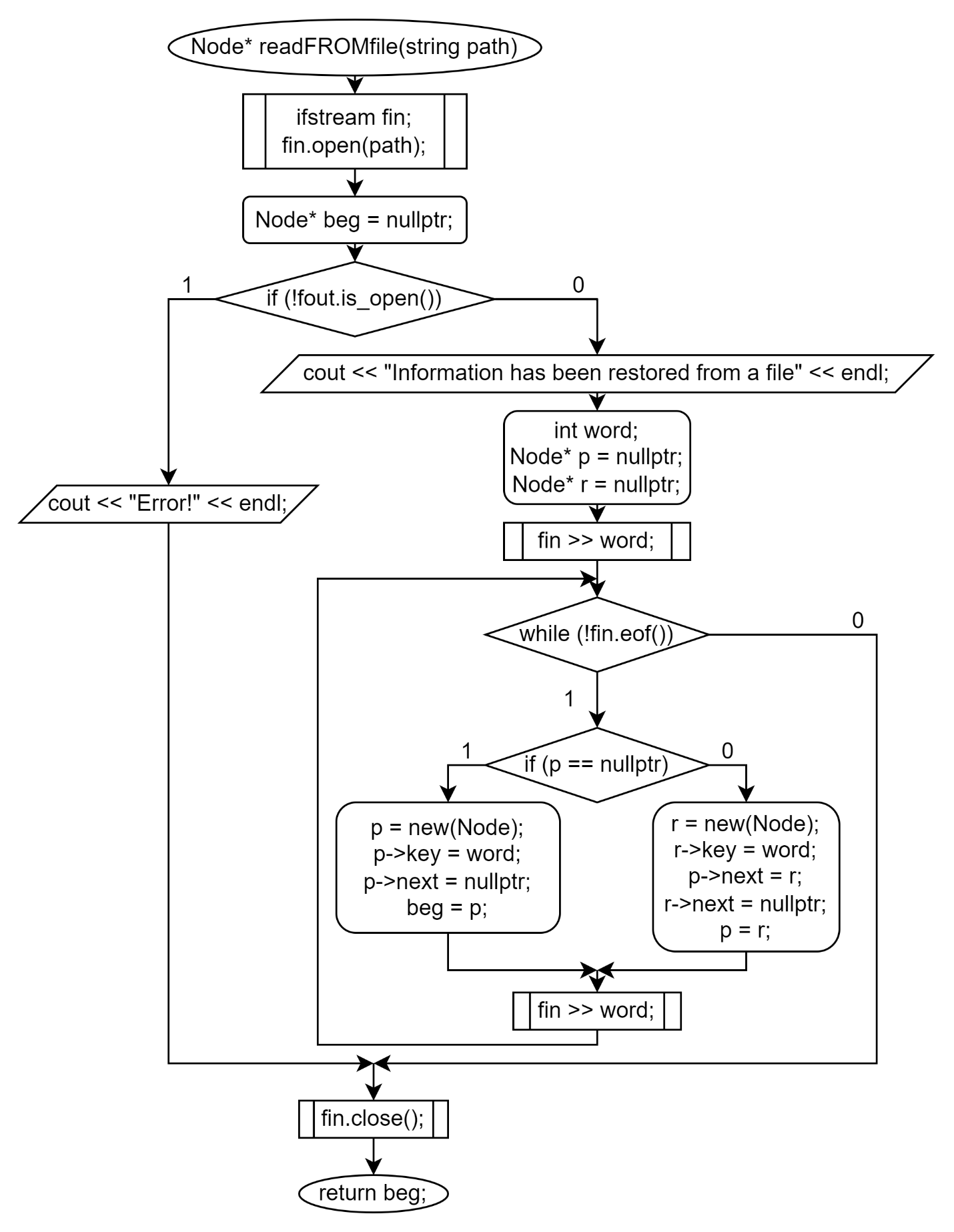


Рис. 7 – Функция Node\* readFROMfile()

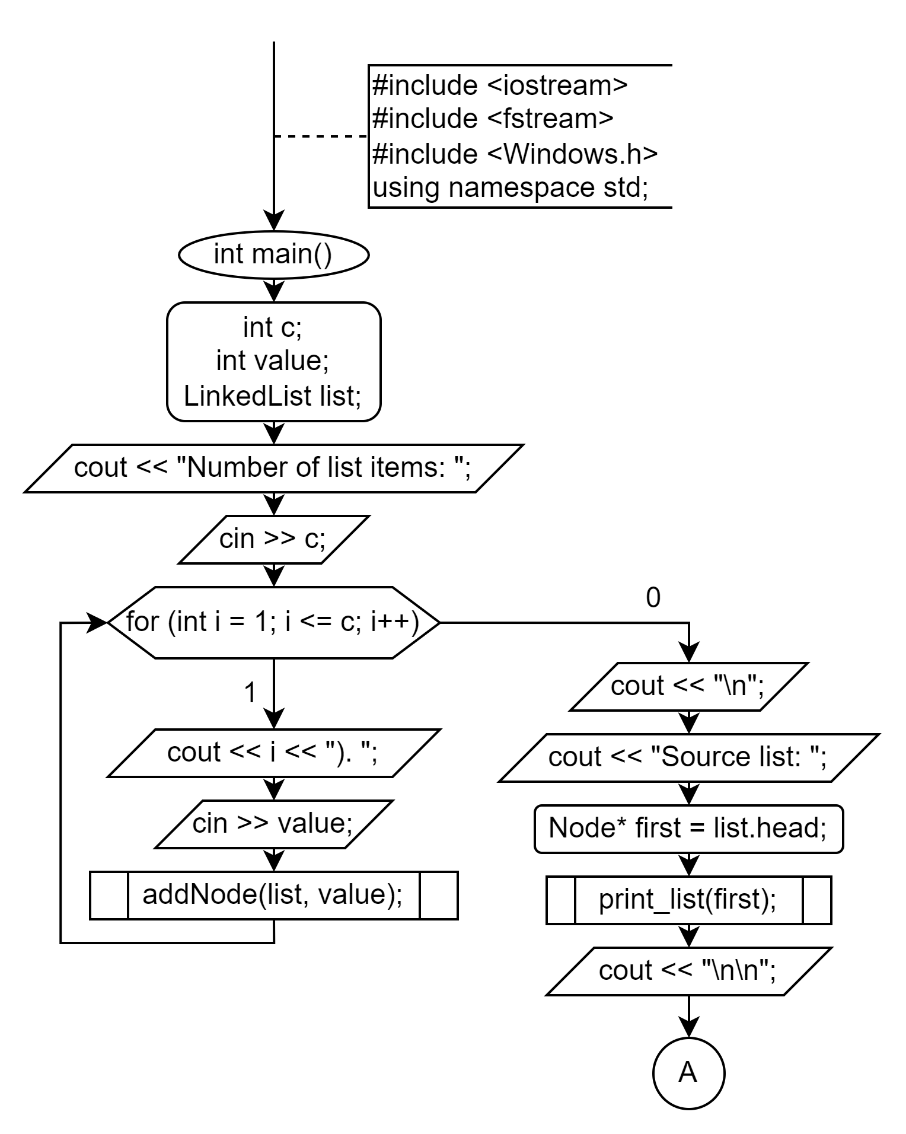


Рис. 8 – Функция int main()

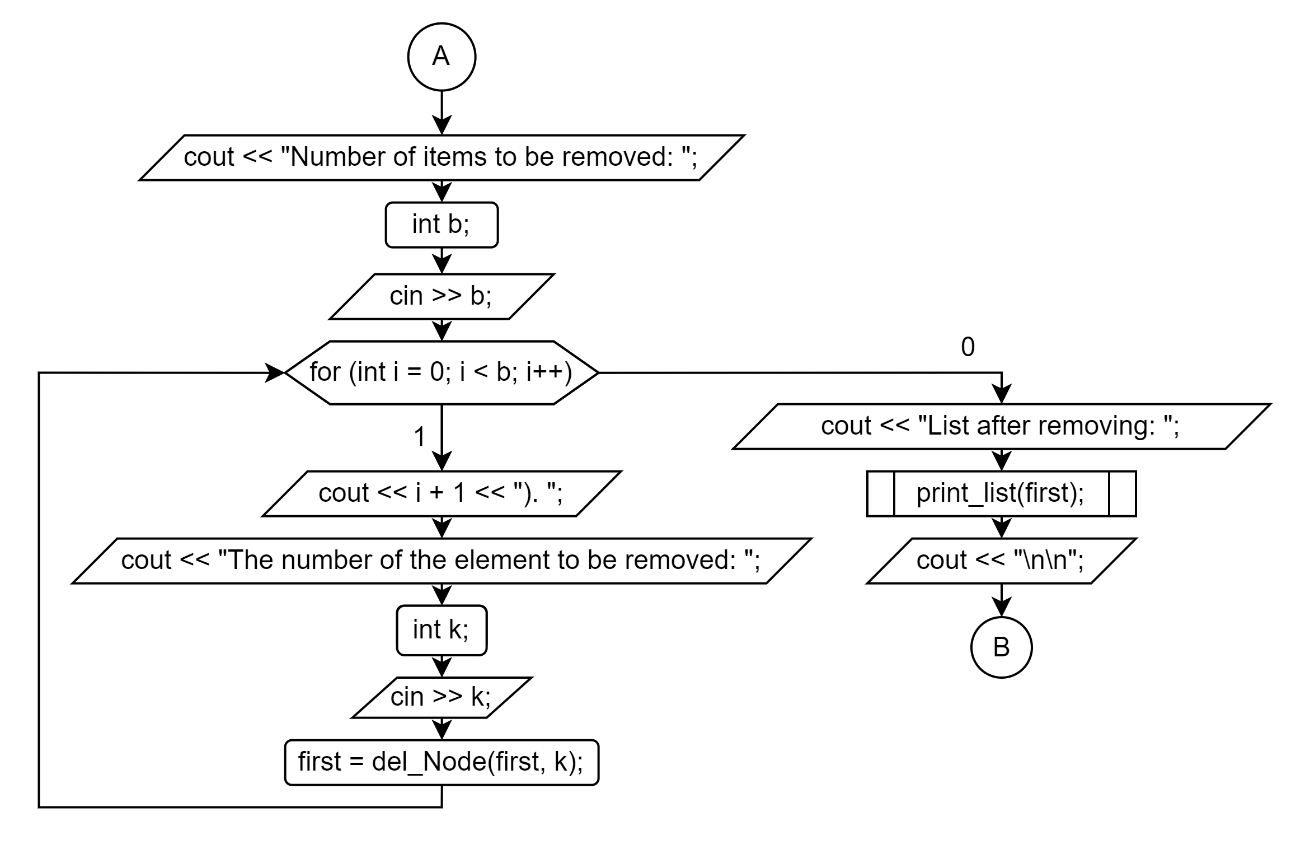


Рис. 9 – Функция int main()

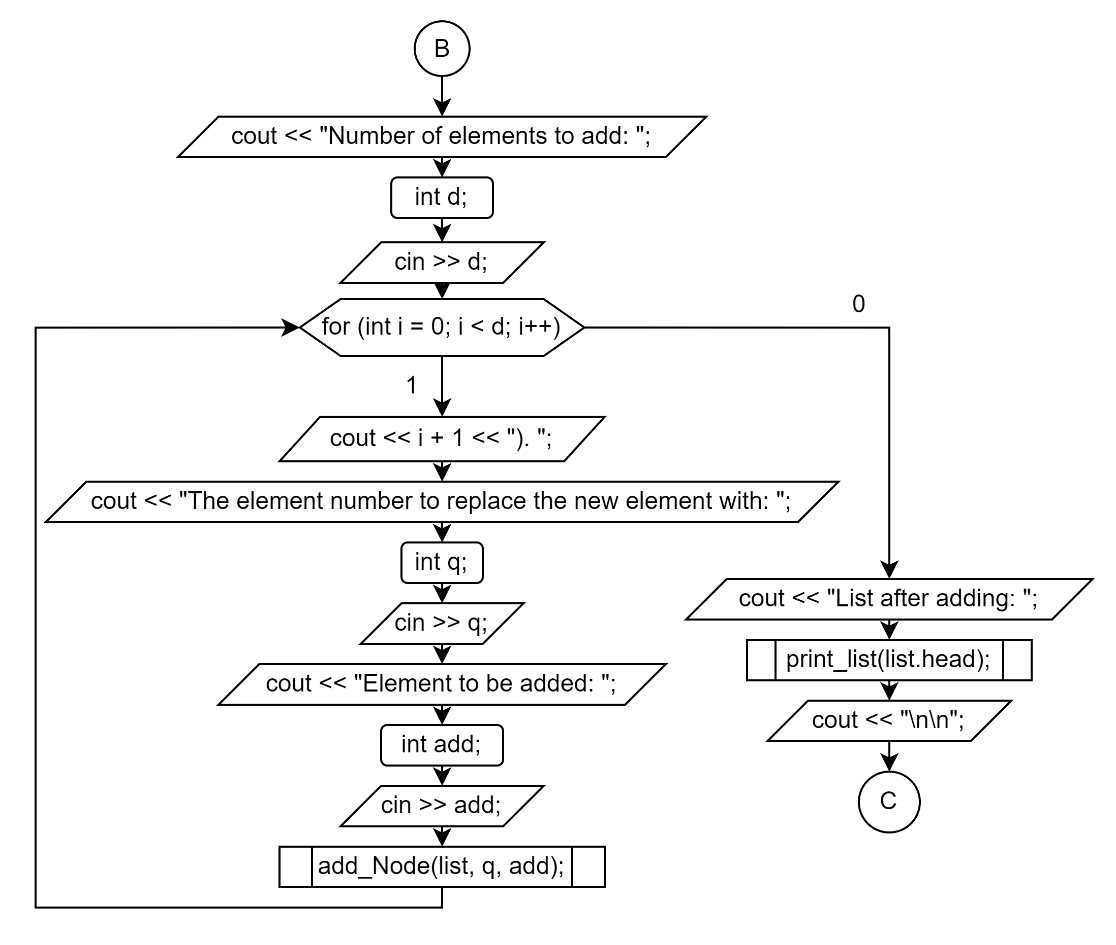


Рис. 10 – Функция int main()

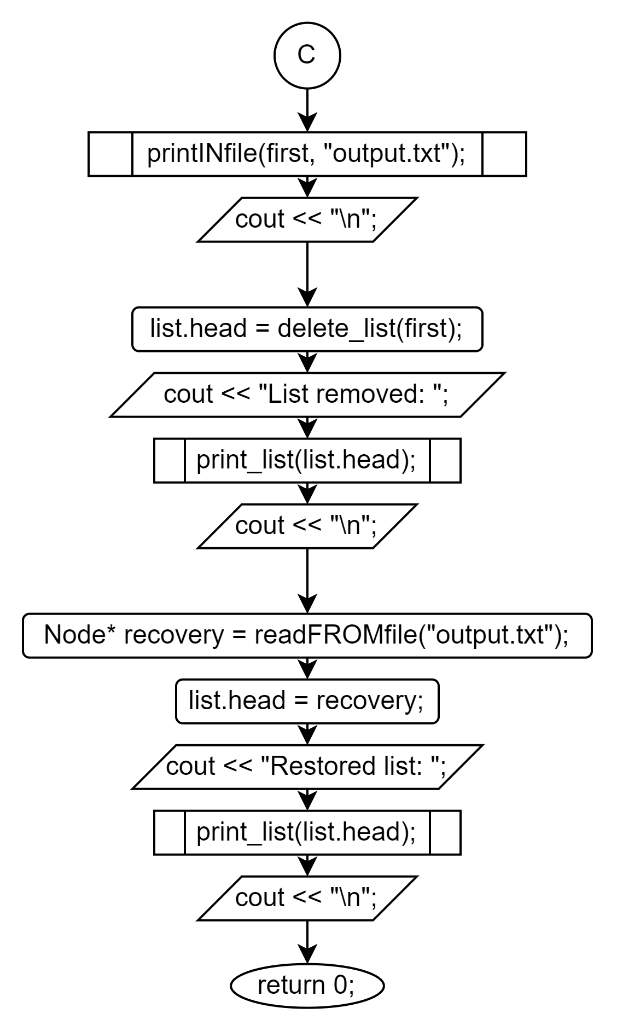


Рис. 11 – Функция int main()



Рис. 12 – Код программы на языке С++

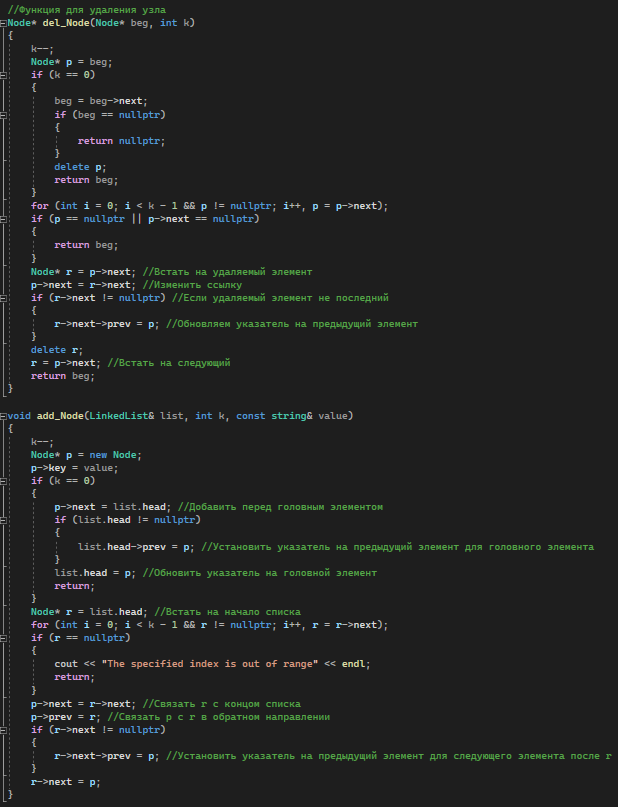


Рис. 13 – Код программы на языке С++

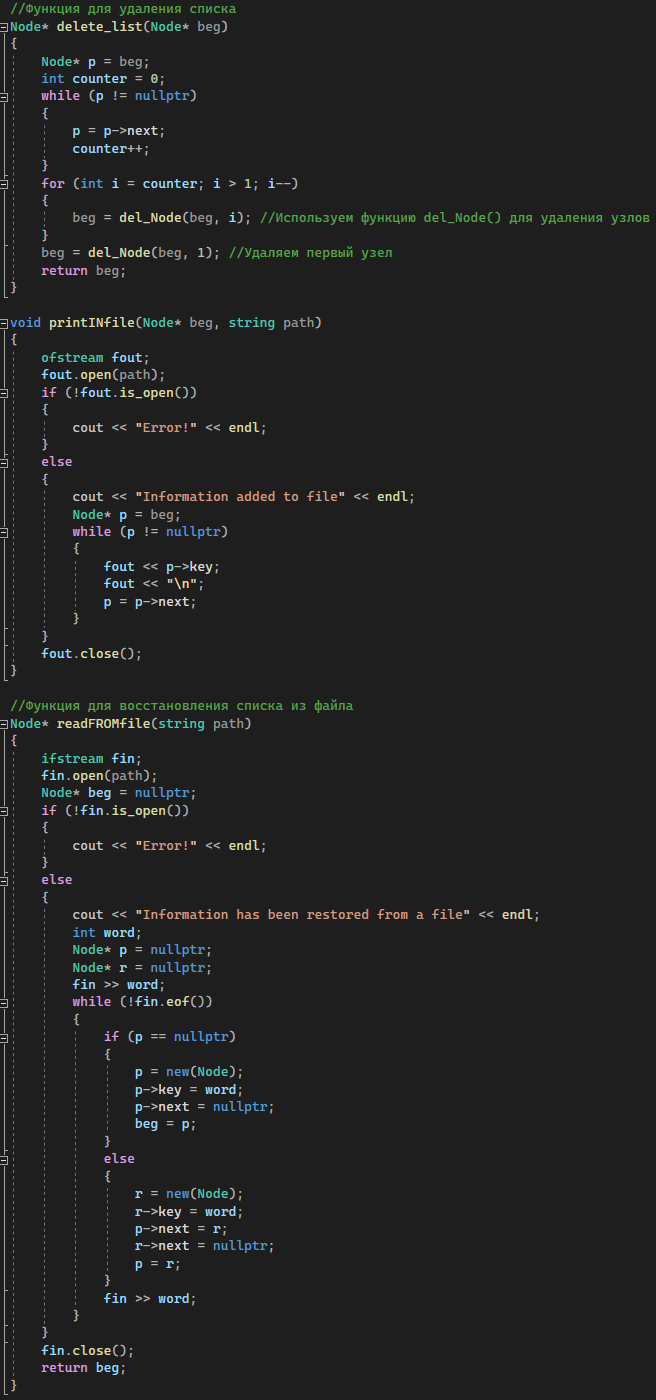


Рис. 14 – Код программы на языке С++

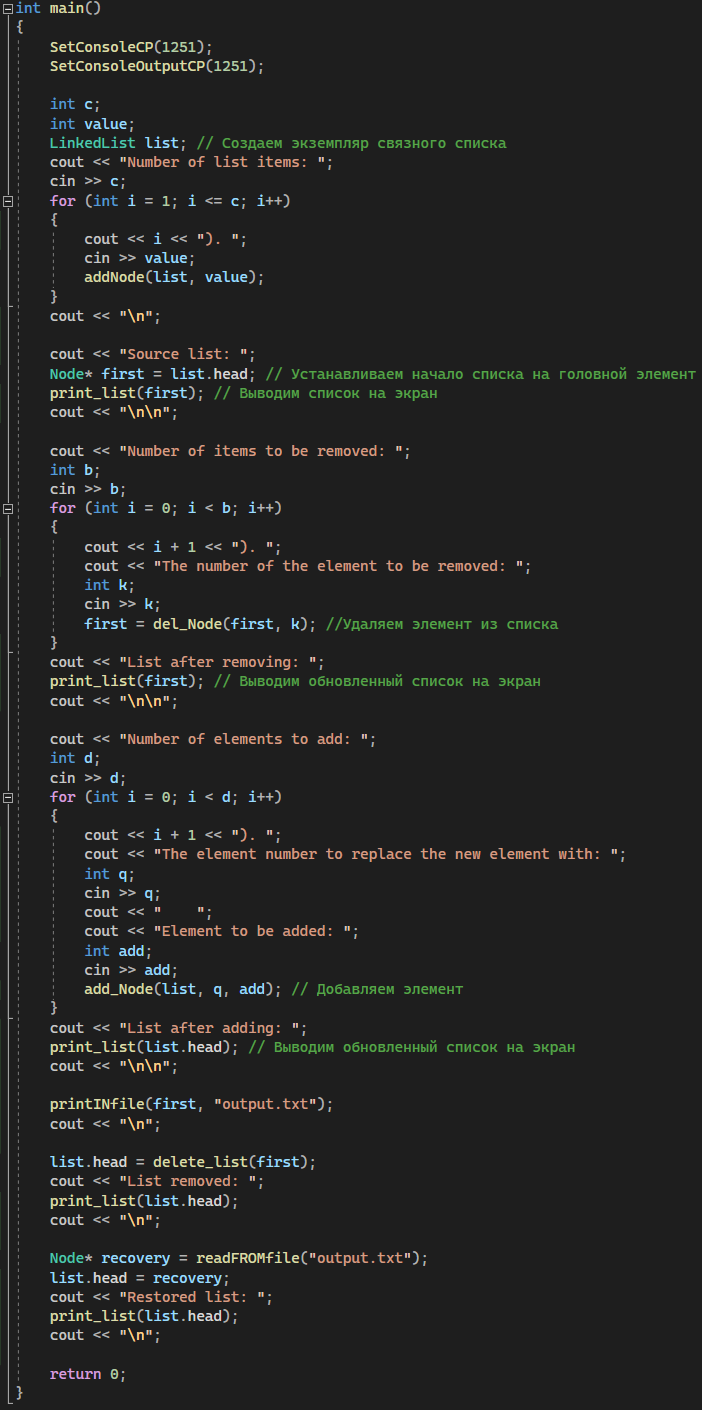


Рис. 15 – Код программы на языке С++

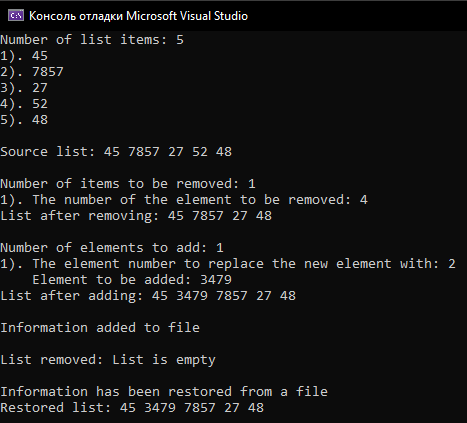


Рис. 16 – Результат работы программы