**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

**Кафедра прикладной информатики и теории вероятностей**

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

Студент: Леснухин Д.Д..

Группа: НПИбд-02-22

**МОСКВА**

2023 г.

**Цель работы:**

Написать компьютерную программу, содержащую

 - Описание структуры, содержащей поля типа string, int, double;

 - Набор функций для работы со списком на базе этой структуры:

     - Добавление элемента в начало списка;

     - Добавление элемента в конец списка;

     - Добавление элемента в список после заданного элемента;

     - Добавление элемента в список перед заданным элементом;

     - Удаление из списка элемента с заданным именем;

     - Вывод содержания списка на экран;

 - Функцию main, содержащую сценарий работы со списком, использующий разработанный инструментарий.

Представить результаты в виде двух файлов:

 - Компьютерная программа на С++;

 - Отчет о выполнении лабораторной работы с описанием алгоритма и структуры программы.

**Алгоритм:**

1. Объявить структуру Node с полями name, age и salary типов string, int и double соответственно, а также указателем на следующий элемент списка.

2. Объявить класс LinkedList с приватным полем head типа Node\*, указывающим на голову списка.

3. Описать в классе метод createNode, который возвращает указатель на новый элемент типа Node со значениями переданных аргументов.

4. Описать в классе конструктор LinkedList, который инициализирует head значением NULL.

5. Описать в классе метод insertAtBeginning, который добавляет новый элемент в начало списка. Если список пуст, то head указывает на новый элемент; иначе новый элемент становится головой списка, а указатель на старую голову сохраняется в поле next нового элемента.

6. Описать в классе метод insertAtEnd, который добавляет новый элемент в конец списка. Если список пуст, то head указывает на новый элемент; иначе последний элемент списка найден, и его поле next указывает на новый элемент.

7. Описать в классе метод insertAfterNode, который добавляет новый элемент после существующего элемента. Если список пуст, выводится сообщение об ошибке. Если указанный элемент не найден в списке, выводится сообщение об ошибке. Иначе новый элемент добавляется после найденного элемента.

8. Описать в классе метод insertBeforeNode, который добавляет новый элемент перед существующим элементом. Если список пуст, выводится сообщение об ошибке. Если указанный элемент является головой списка, новый элемент становится головой списка. Если указанный элемент не найден в списке, выводится сообщение об ошибке. Иначе новый элемент добавляется перед найденным элементом.

9. Описать в классе метод deleteNode, который удаляет элемент с заданным именем. Если список пуст, выводится сообщение об ошибке. Если указанный элемент является головой списка, голова списка переназначается. Если указанный элемент не найден в списке, выводится сообщение об ошибке. Иначе удаляется найденный элемент.

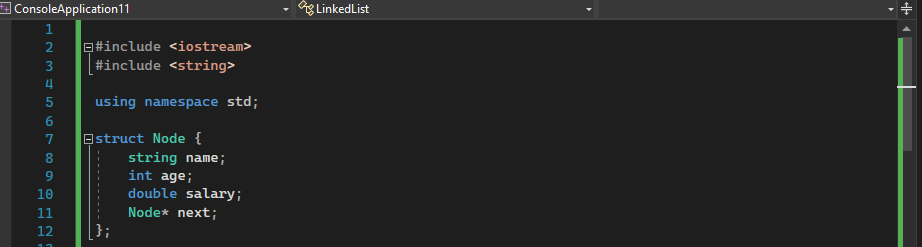
10. Описать в классе метод printList, который выводит содержимое списка на экран. Если список пуст, выводится сообщение об ошибке.

11. В функции main объявить объект класса LinkedList.

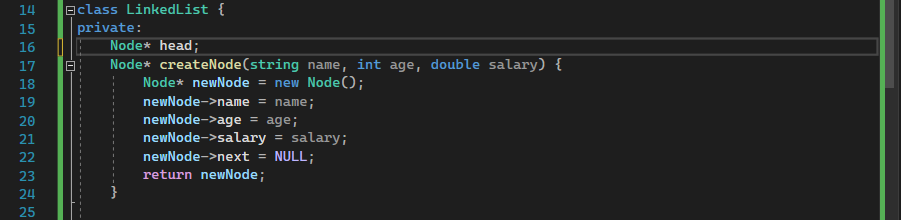
12. Использовать методы класса LinkedList для добавления, удаления и вывода элементов списка.

**Структура программы:**

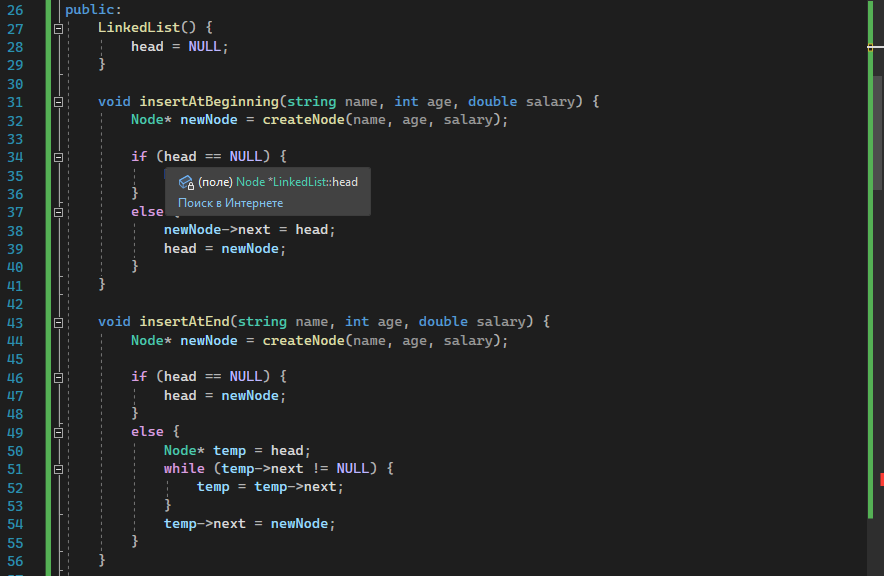
1. Описание структуры Node с полями name, age и salary типов string, int и double соответственно, а также указателем на следующий элемент списка.



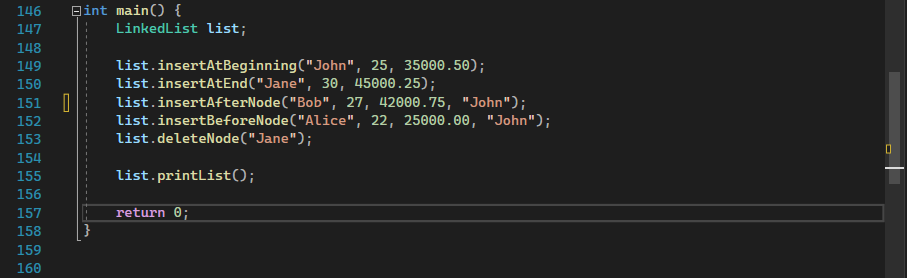
1. Описание класса LinkedList с приватным полем head типа Node\*, указывающим на голову списка, и публичными методами insertAtBeginning, insertAtEnd, insertAfterNode, insertBeforeNode, deleteNode и printList.



1. Описание вспомогательной функции createNode, создающей новый элемент списка и возвращающей его указатель.

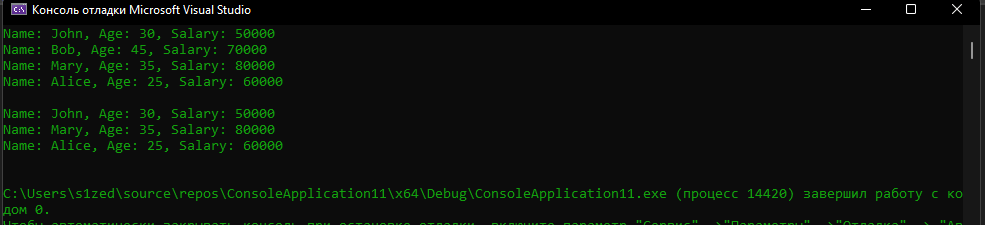


1. В функции main создается объект класса LinkedList.



5. С использованием методов класса LinkedList создается список, производится удаление элемента и вывод на экран.

Пример вывода программы:



**Вывод:**

Написал компьютерную программу, содержащую

 - Описание структуры, содержащей поля типа string, int, double;

 - Набор функций для работы со списком на базе этой структуры:

     - Добавление элемента в начало списка;

     - Добавление элемента в конец списка;

     - Добавление элемента в список после заданного элемента;

     - Добавление элемента в список перед заданным элементом;

     - Удаление из списка элемента с заданным именем;

     - Вывод содержания списка на экран;

 - Функцию main, содержащую сценарий работы со списком, использующий разработанный инструментарий.