МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА» (БГТУ им. В.Г. Шухова)

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №5

по дисциплине: «ООП»

Выполнил: ст. группы ПВ-211

Чувилко Илья Романович

Проверил:

Буханов Дмитрий Геннадьевич

Харитонов Сергей Дмитриевич

Белгород 2023 г.

Тема: Классы, виды отношений. Наследование.

Вариант по списку: 23

Цель работы: получение теоретических знаний в области разработки классов, получение практических навыков реализаций классов и отношений между ними.

Задание к работе: В соответствии с вариантом (((номер по списку + 5)%10)+1) выполнить построение объектной модели (использовать не менее 5 объектов) заданной предметной области (задание 1), разработать диаграмму классов для описанной объектной модели (не менее 7 классов), и реализовать предложенные классы (задание 2).

Вариант: 9 (((23 + 5)%10)+1)

Задание 1: Система кредитования граждан.

Диаграмма объектов:

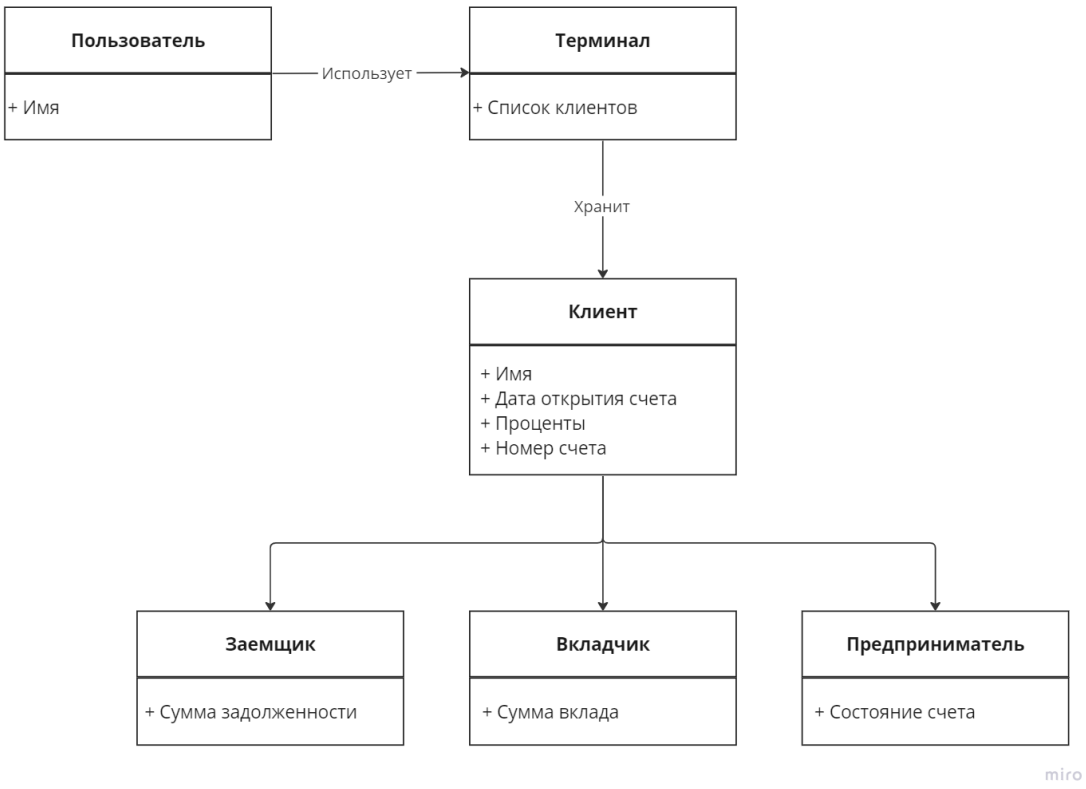
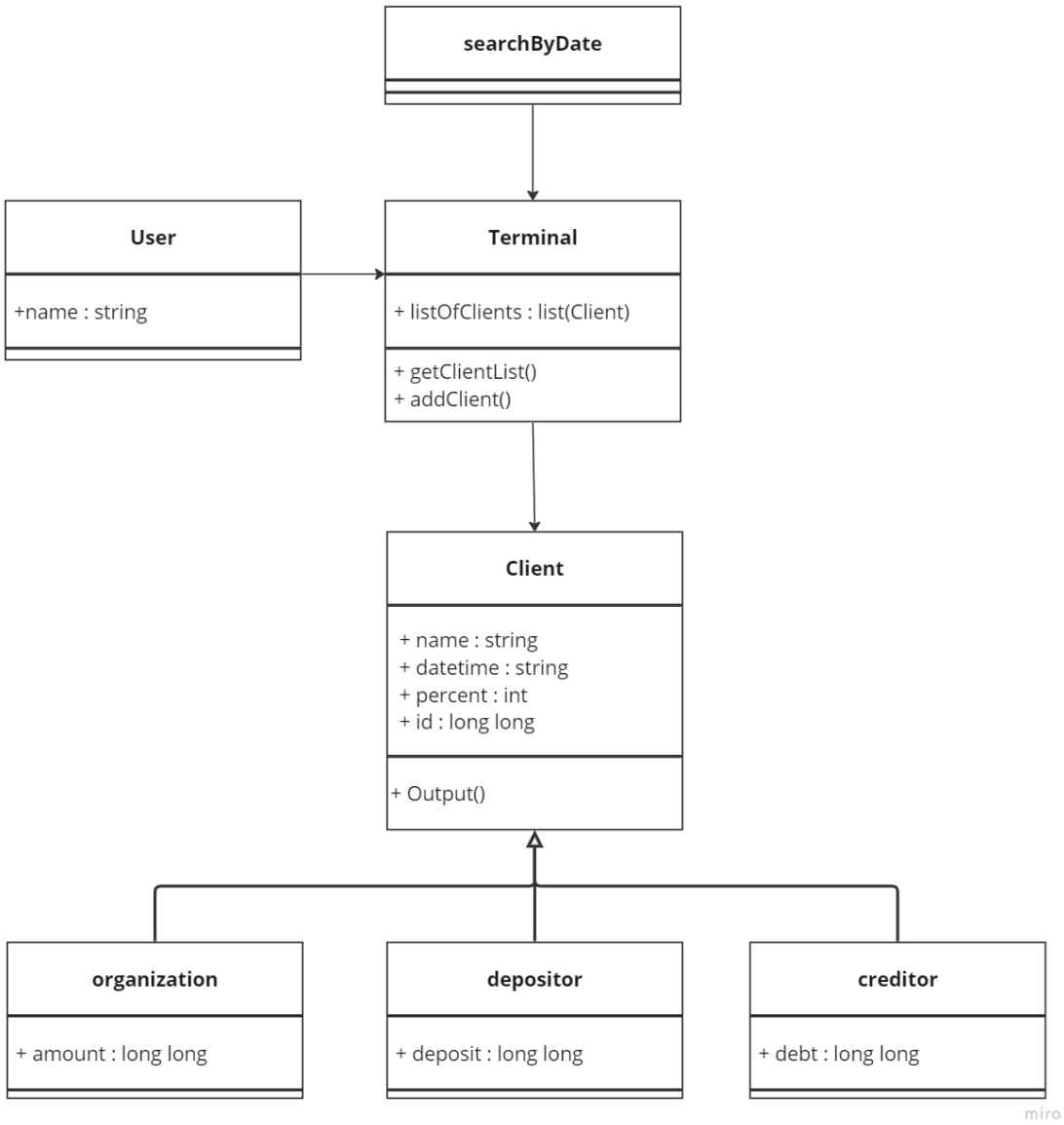


Диаграмма классов:



Задание 2

Задание 1. Создать абстрактный класс Клиент с методами, позволяющими вывести на экран информацию о клиентах банка, а также определить соответствие клиента критерию поиска.

```
class Client {
  protected:
  string name;
  string datetime;
  int percent;
  long long id;
  public:
  virtual void outputInfo() {
    cout << "Имя: " << name;
    cout << "\nДата открытия счета: " << datetime;
    cout << "\nПроцентная ставка: " << percent;
    cout << "\nУникальный идентификатор: " << id;
  }
};
```

Задание 2. Создать производные классы: Вкладчик (фамилия, дата открытия вклада, размер вклада, процент по вкладу), Кредитор (фамилия, дата выдачи кредита, размер кредита, процент по кредиту, остаток долга), Организация (название, дата открытия счёта, номер счёта, сумма на счету) со своими методами вывода информации на экран, и определения соответствия дате (открытия вклада, выдаче кредита, открытия счёта)

```
class entrepreneur : public Client {
 private:
 long long amount;
 public:
 entrepreneur(string name, string datetime, int percent, long long id,
              long long amount) {
  this->name = name;
  this->datetime = datetime;
  this->percent = percent;
  this->id = id;
  this->datetime = datetime;
  this->amount = amount;
 }
 void outputInfo() override {
  Client::outputInfo();
  cout << "\nСостояние счета: " << amount<< '\n';
 }
};

class depositor : public Client {
 private:
 unsigned long long deposit;
 public:
 depositor(string name, string datetime, int percent, long long id,
           long long deposit) {
  this->name = name;
  this->datetime = datetime;
  this->percent = percent;
  this->id = id;
  this->deposit = deposit;
 }
 void outputInfo() override {
  Client::outputInfo();
  cout << "\nДепозит: " << deposit << '\n';
 }
};

class creditor : public Client {
 private:
 unsigned long long debt;
 public:
 creditor(string name, string datetime, int percent, long long id,
           long long debt) {
  this->name = name;
  this->datetime = datetime;
  this->percent = percent;
  this->id = id;
  this->debt = debt;
 }
 void outputInfo() override {
  Client::outputInfo();
  cout << "\nЗадолженность: " << debt << '\n';
 }
};
```

Задание 3. Создать базу (массив) из n клиентов, вывести полную информацию из базы на экран, а также организовать поиск клиентов, начавших сотрудничать с банком в заданную дату.

```
class Terminal : public Client {
  private:
  vector<Client> data;
  public:
  vector<Client> *getClientList() {
    return &data;
  }
  void addClient(Client c) {
    data.push_back(c);
  }
  void searchByDate(string date) {
    for (auto &i: data)
      if (i.datetime == date)
        i.outputInfo();
  }
};



int main() {
  SetConsoleOutputCP(CP_UTF8);
  depositor vladimir = {"Владимир", "20.10.1999", 5, 0, 5000};
  depositor daniil = {"Даниил",   "01.01.2007", 8, 1, 312341};
  creditor luda = {"Людмила", "08.07.2022", 10,  2, 10000};
  creditor dima = {"Дмитрий", "01.03.2023", 11,  3, 312341};
  entrepreneur bistro = {"Бистро", "01.03.2023", 3,  4, 1231231};
  entrepreneur babula = {"ООО \"Как у бабушки\"", "01.03.2023", 3,  5, 58378645734};
  Terminal t;
  t.addClient(vladimir);
  t.addClient(daniil);
  t.addClient(luda);
  t.addClient(dima);
  t.addClient(bistro);
  t.addClient(babula);

  cout << "\n\n Поиск по дате: 01.03.2023\n";
  t.searchByDate("01.03.2023");
  return 0;
}
```

Вывод: в ходе лабораторной работы мы получили теоретические знания в области разработки классов, получили практические навыков реализации классов и отношений между ними