Для выполнения данной лабораторной работы потребуется:

1. **Построение модульной структуры проекта** в виде диаграммы с использованием предложенных в задании модулей:
   * Головной модуль (Main module);
   * Модуль управления устройством считывания кредитной карты (Credit card control module);
   * Модуль аутентификации (Authentication module);
   * Модуль получения и обработки запроса на обслуживание (Reception and processing module);
   * Модуль данных кредитной карты (Credit card data module).
2. **Создание спецификаций для каждого модуля**, включающих:
   * Синтаксическую спецификацию (описание входов/выходов, параметров и возвращаемых значений);
   * Функциональную спецификацию (описание задач модуля и их выполнения).
3. **Построение структурной карты проекта** в виде дерева, где узлы обозначают модули, а дуги – связь между ними.

**Ход работы**

**1. Построение модульной структуры**

Модульная структура должна быть представлена в виде дерева. Главный модуль (Main module) находится на вершине структуры и управляет всеми остальными модулями. Связи между модулями показывают последовательность их вызова и обмен данными.

Диаграмма (пример):

mathematica

Копировать код

Main Module

│

├── Credit Card Control Module

│

├── Authentication Module

│

└── Reception and Processing Module

└── Credit Card Data Module

**2. Спецификации модулей**

**2.1. Main Module**

* **Синтаксическая спецификация**:
  + Вход: initAction(action: string): void
  + Выход: result: bool
* **Функциональная спецификация**:
  + Инициализация процесса работы с банкоматом.
  + Управление вызовами других модулей (ввод карты, проверка пароля, обработка запроса, завершение работы).

**2.2. Credit Card Control Module**

* **Синтаксическая спецификация**:
  + Вход: readCardData(cardData: byte[]): void
  + Выход: cardInfo: {number: string, owner: string, expiry: date}
* **Функциональная спецификация**:
  + Обеспечение ввода, считывания и удаления кредитной карты.

**2.3. Authentication Module**

* **Синтаксическая спецификация**:
  + Вход: verifyPin(pin: string): bool
  + Выход: authFlag: bool
* **Функциональная спецификация**:
  + Проверка введенного PIN-кода.
  + Передача результата проверки в главный модуль.

**2.4. Reception and Processing Module**

* **Синтаксическая спецификация**:
  + Вход: processRequest(requestType: string, amount: float): string
  + Выход: result: string
* **Функциональная спецификация**:
  + Обработка операций клиента, таких как запрос баланса, выдача наличных, распечатка чеков.

**2.5. Credit Card Data Module**

* **Синтаксическая спецификация**:
  + Вход: fetchData(cardNumber: string): {balance: float, transactions: list}
  + Выход: data: object
* **Функциональная спецификация**:
  + Хранение и предоставление данных о кредитной карте.

**3. Построение структурной карты**

Для построения диаграммы можно использовать такие инструменты, как **EasyCASE Professional Version 4.21.016**, Visio или аналогичные.

**Отчет**

Отчет должен включать:

1. **Название работы**: Лабораторная работа №4. Разработка модульной структуры проекта.
2. **Цель работы**: Изучение процесса разработки модульной структуры программного обеспечения с использованием структурных карт.
3. **Ход работы**:
   * Построение модульной структуры (текст и диаграмма).
   * Спецификации модулей.
4. **Выводы**: Например:
   * Модульная структура позволяет организовать проект более структурированно.
   * Использование диаграмм упрощает проектирование и тестирование системы.
5. **Ответы на контрольные вопросы**:
   * **1. Цель разработки модульной структуры**: Повышение надежности, уменьшение сцепления и упрощение тестирования системы.
   * **2. Понятие программного модуля**: Обособленная часть программы, реализующая определенную функциональность. Связь между модулями осуществляется через параметры или общую область памяти.
   * **3. Виды связности модулей**: По данным, по управлению, логическая.
   * **4. Виды целостности модулей**: Функциональная, логическая, временная, процедурная.
   * **5. Типовые модульные структуры**: Иерархическая, линейная, блочная.
   * **6. Проектирование модульной структуры**: С помощью диаграмм, структурных карт, обеспечивая минимальное сцепление и максимальную связность модулей.
   * **7. EasyCASE**: Программа для создания диаграмм и карт модульной структуры.