

СТРУКТУРНО-МОДУЛЬНОЕ ПРОЕКТИРОВАНИЕ

Парадигма (от греческого *παράδειγμα* – пример, модель, образец) – в философии, социологии исходная концептуальная схема, модель постановки проблем и их решения, методов исследования, господствующих в течение определенного исторического периода в научном сообществе. Смена парадигм представляет собой научную революцию или эволюционный переход.

Парадигма программирования – это парадигма, определяющая некоторый цельный набор идей и рекомендаций, формирующих стиль и технику написания программ. Например, в объектно-ориентированном программировании программист рассматривает программу как набор взаимодействующих объектов, тогда как в функциональном программировании программа представляется в виде цепочки вычисления функций.

Для достижения тех или иных критериев существуют различные приемы написания кода, которые иногда могут противоречить друг другу. Вместе с тем имеется ряд рекомендаций, которые должны соблюдаться при написании любой программы независимо от избранной парадигмы программирования. К ним относятся: модульность, функциональная и структурная декомпозиция, отсутствие прямых связей через глобальные переменные и др.

Отличие парадигм программирования проявляется в разнообразии реализаций моделей состояния и поведения, а также отношений между этими понятиями, осуществляемых через такие элементарные программные объекты, как данные и операции.

В *процедурном подходе* абстрагирование от конкретных экземпляров достигается за счет введения понятий «абстрактный тип данных» и «процедура» (понятие «функция» используется как синоним процедуры).

В *объектно-ориентированном* программировании «базовыми кирпичиками» являются класс и метод класса.

СТРУКТУРНО-МОДУЛЬНОЕ ПРОЕКТИРОВАНИЕ

Основные подходы при программировании:

- Нисходящее проектирование;
- Модульное программирование;
- Структурное программирование.

НИСХОДЯЩЕЕ ПРОЕКТИРОВАНИЕ

Метод нисходящего проектирования предполагает последовательное разложение общей функции обработки данных на простые функциональные элементы ("сверху-вниз").

В результате строится иерархическая схема, отражающая состав и взаимоподчиненность отдельных функций, которая носит название функциональная структура алгоритма (ФСА) приложения.

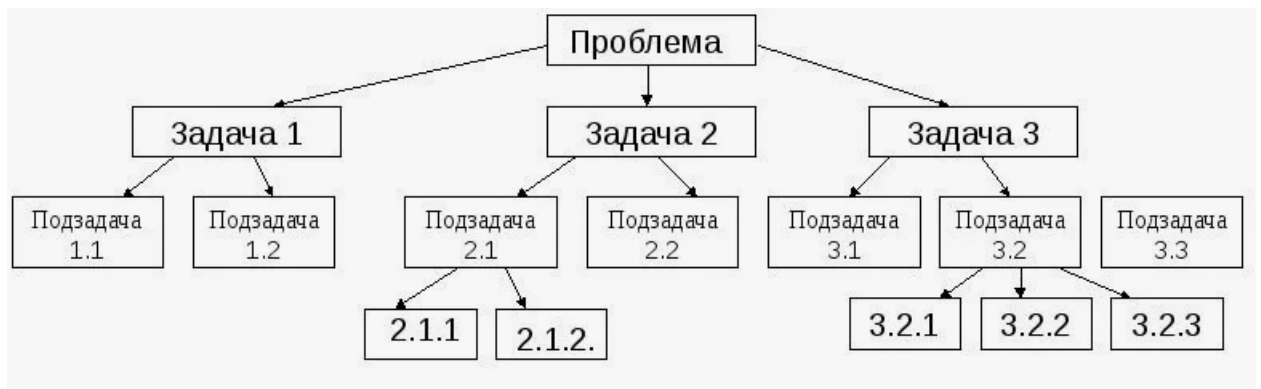


Рис.1.

Подобная структура приложения (Рис. 1) отражает наиболее важное - состав и взаимосвязь функций обработки информации для реализации приложений, хотя и не раскрывает логику выполнения каждой отдельной функции, условия или периодичность их вызовов.

МОДУЛЬНОЕ ПРОГРАММИРОВАНИЕ

Модульное программирование основано на понятии модуля - логически взаимосвязанной совокупности функциональных элементов, оформленных в виде отдельных программных модулей.

Модуль характеризуют:

- один вход и один выход - на входе программный модуль получает определенный набор исходных данных, выполняет содержательную обработку и возвращает один набор результатных данных, т.е. реализуется стандартный принцип IPO (Input - Process - Output) - вход-процесс-выход;
- функциональная завершенность - модуль выполняет перечень регламентированных операций для реализации каждой отдельной функции в полном составе, достаточных для завершения начатой обработки;
- логическая независимость - результат работы программного модуля зависит только от исходных данных, но не зависит от работы других модулей;
- слабые информационные связи с другими программными модулями - обмен информацией между модулями должен быть по возможности минимизирован;
- обозримый по размеру и сложности программный элемент.

Каждый модуль состоит из спецификации и тела. Спецификации определяют правила использования модуля, а тело - способ реализации процесса обработки.

Модульная структура программных продуктов

Принципы модульного программирования программных продуктов во многом сходны с принципами нисходящего проектирования. Сначала определяются состав и подчиненность функций, а затем - набор программных модулей, реализующих эти функции (Рис.2.).

Однотипные функции реализуются одними и теми же модулями. Функция верхнего уровня обеспечивается главным модулем; он управляет выполнением нижестоящих функций, которым соответствуют подчиненные модули.

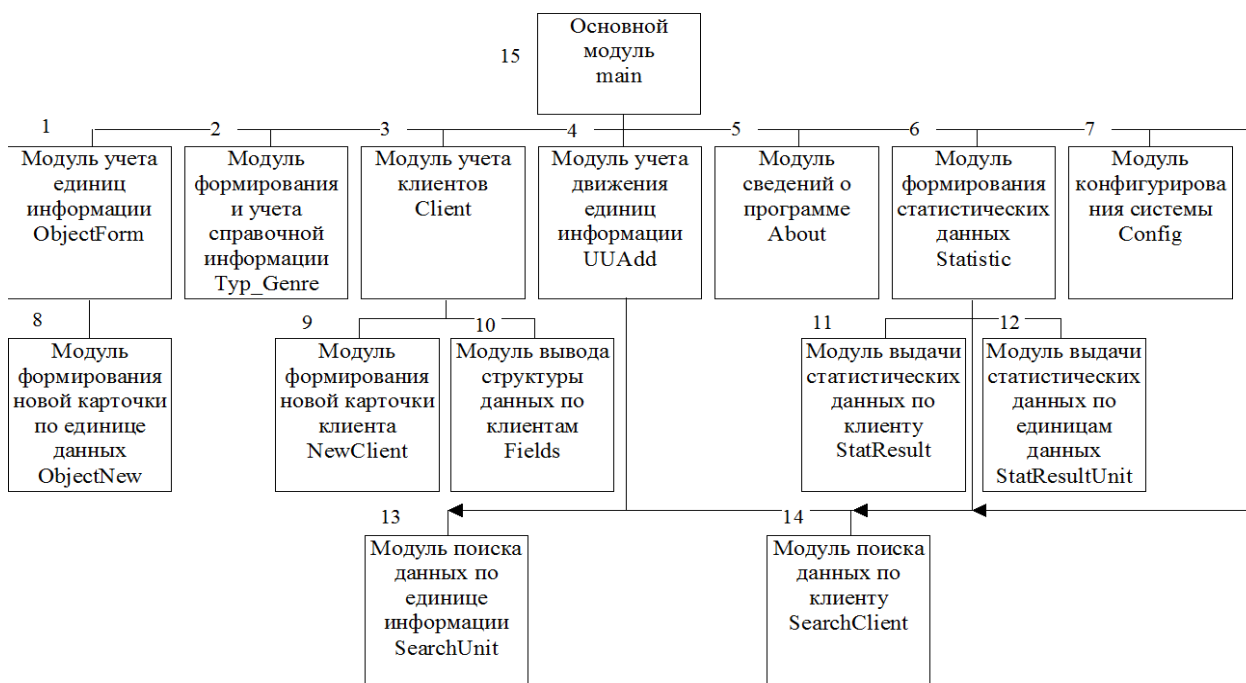


Рис.2. Функционально-модульная структура приложения

HIPO-СХЕМЫ

Технология HIPO (Hierarchical Input Process Output Diagrams) - многоуровневая технология проектирования и документирования программных моделей, основанная на использовании системы шаблонов, бланков и типовых диаграмм. HIPO-диаграммы (Hierarchy Input-Processing-Output)

Данный метод используется после подготовки структурной схемы модулей и направлен на доработку взаимодействия отдельных элементов модуля. При этом рассматриваются входные и выходные данные, а также процессы их преобразования. Общий вид данных схем можно представить так:

Исходное описание Системы Физика Схема Программа Номер: 1.1	Разработать электронный учебник по теме «Законы идеального газа и уравнение состояния»	Лист: 1 Автор: Васильцов А. Отдел: ИУК4-216 Дата: 01.03.2023
--	--	---

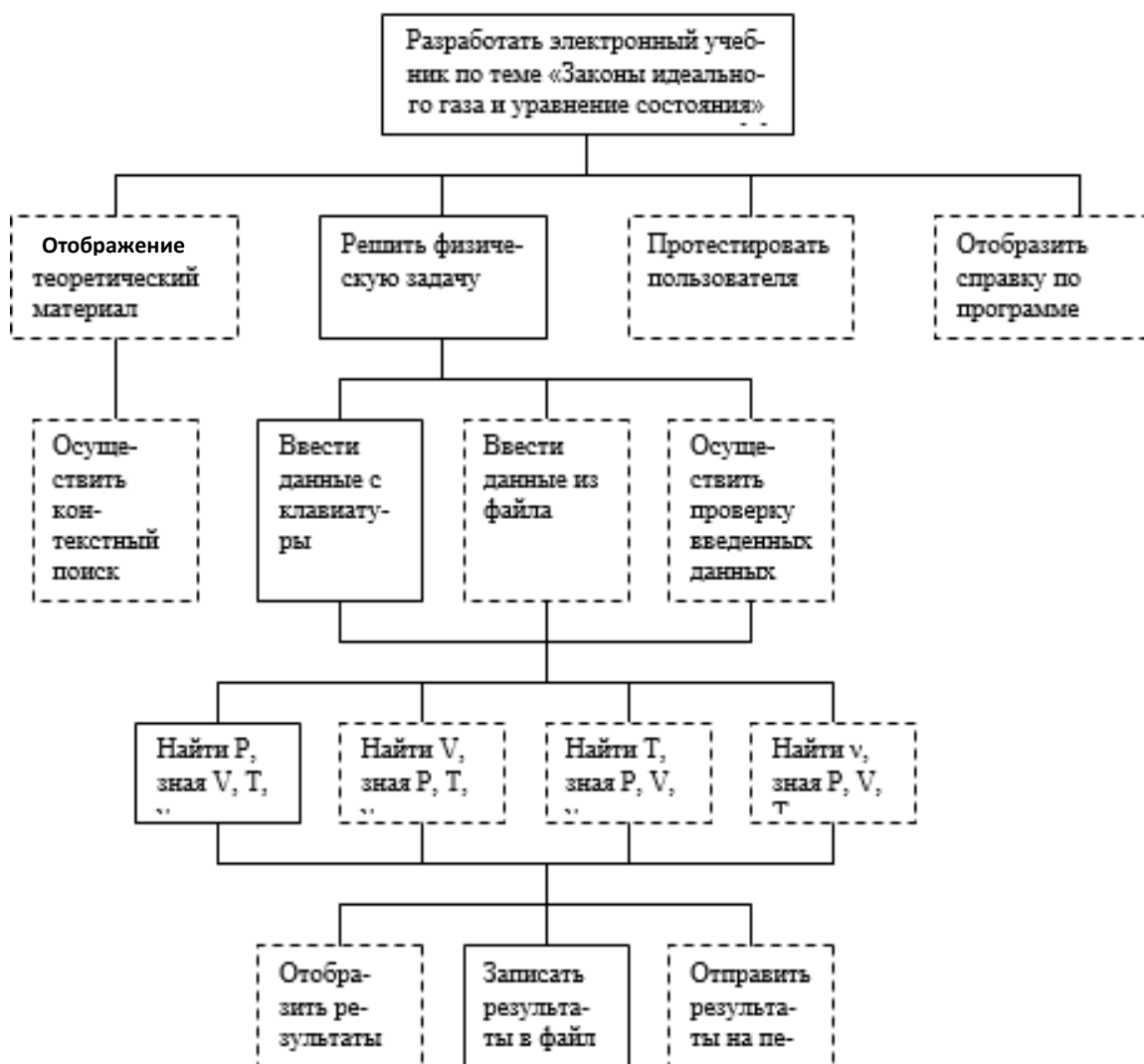


Рис.3. Схема программы

СТРУКТУРНОЕ ПРОГРАММИРОВАНИЕ

Структурное программирование основано на модульной структуре программного продукта и типовых управляющих структурах алгоритмов обработки данных различных программных модулей (рис. 4).

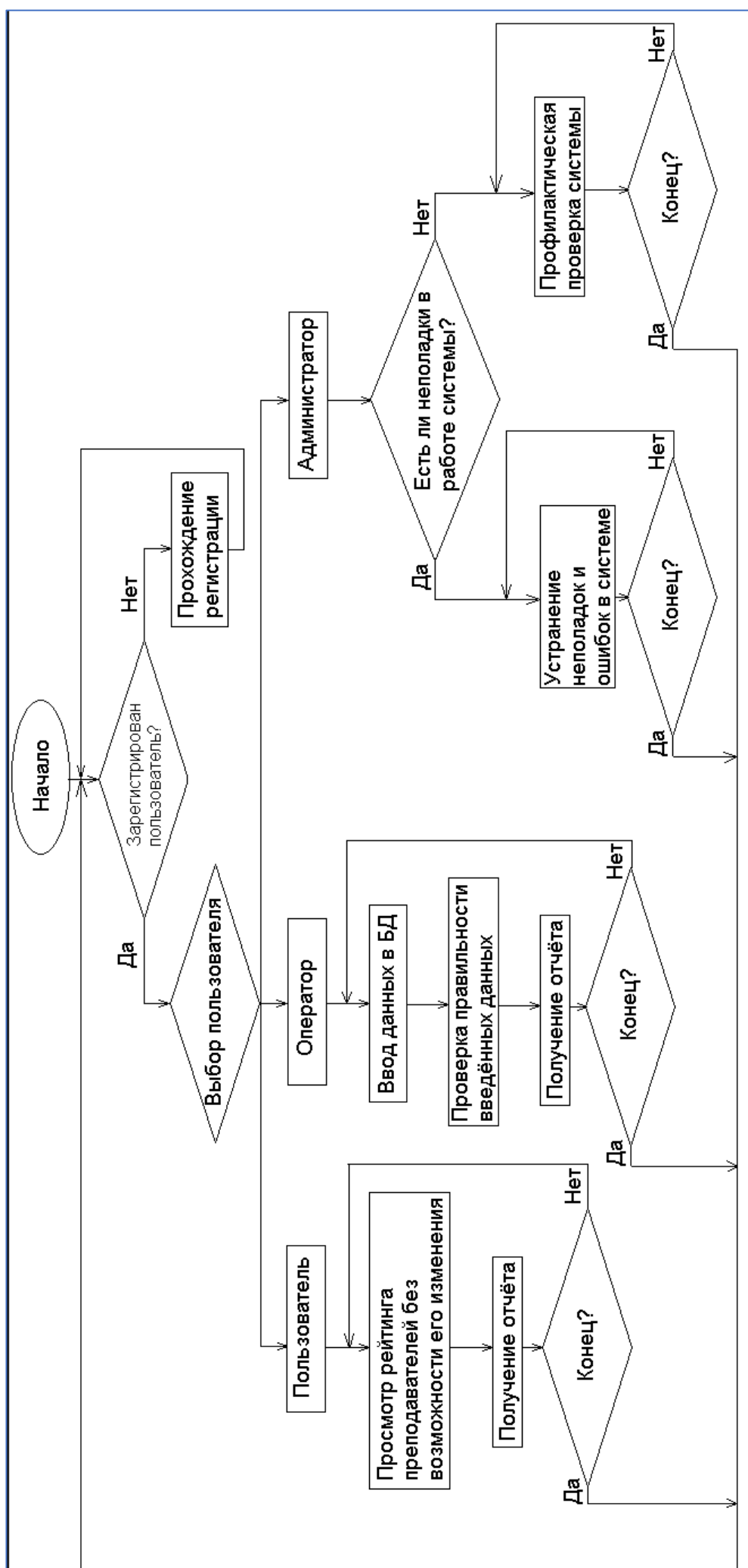


Рис. 4. Блок-схема алгоритма работы в базе данных.

ЗАДАНИЕ

Команда знакомится с постановкой задачи, выделяет модули, которые могут быть разработаны независимо, продумывает их интерфейсную часть.

На основании исходных данных, представленных в техническом задании, провести структурно-модульное проектирование и разработать НПРО-схему создаваемого программного продукта.