



**Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)**

**ФАКУЛЬТЕТ** Информатика и системы управления (ИУ)

**КАФЕДРА** «Информационная безопасность» (ИУ8)

## **ПРОГРАММНЫЙ СИМУЛЯТОР RDP-11**

### **Описание программы**

**А.В.00001-01 13 01**

**листов 12**

Исполнитель, студент группы ИУ8-71

\_\_\_\_\_ Тимощук А.А.  
«\_\_\_» \_\_\_\_\_ 20\_\_ г.

Исполнитель, студент группы ИУ8-71

\_\_\_\_\_ Шаповалов М. Е,  
«\_\_\_» \_\_\_\_\_ 20\_\_ г.

Исполнитель, студент группы ИУ8-71

\_\_\_\_\_ Штырков В. С.  
«\_\_\_» \_\_\_\_\_ 20\_\_ г.

Руководитель курсового проекта,  
преподаватель кафедры ИУ8

\_\_\_\_\_ Рафиков А. Г.  
«\_\_\_» \_\_\_\_\_ 20\_\_ г.

Заведующий кафедрой ИУ8

\_\_\_\_\_ Басараб М. А.  
«\_\_\_» \_\_\_\_\_ 20\_\_ г.

## **Аннотация**

В данном программном документе приведено описание программы “Программный симулятор PDP-11”, предназначенной для изучения архитектуры ЭВМ PDP-11, языка ассемблера, работы внешних устройств.

## Содержание

Аннотация.....	2
Основная часть.....	4
1 Общие сведения.....	4
1.1 Обозначение и наименование программы.....	4
1.2 Программное обеспечение, необходимое для функционирования программы.....	4
1.3 Языки программирования, на которых написана программа.....	4
2 Функциональное назначение.....	4
2.1 Назначение программы.....	4
2.2 Сведения о функциональных ограничениях на применение.....	5
3 Описание логической структуры.....	5
3.1 Алгоритм программы.....	5
3.2 Используемые методы.....	5
3.3 Структура программы с описанием функций составных частей и связи между ними.....	5
3.4 Связи программы с другими программами.....	6
4 Используемые технические средства.....	6
5 Вызов и загрузка.....	6
6 Входные данные.....	6
7 Выходные данные.....	7
Лист регистрации изменений.....	8
Приложения.....	9

## **Основная часть**

### **1 Общие сведения**

#### **1.1 Обозначение и наименование программы**

Наименование: «Программный симулятор PDP-11».

#### **1.2 Программное обеспечение, необходимое для функционирования программы**

Для функционирования программы необходим компьютер под управлением ОС Windows, или Linux, или MacOS.

Полный список поддерживаемых ОС и необходимые библиотеки для запуска представлен на следующей странице:  
<https://github.com/dotnet/core/blob/main/release-notes/6.0/supported-os.md>

#### **1.3 Языки программирования, на которых написана программа**

Исходным языком программирования для программы является C#. Версия .NET – 6.0. Для написания графического интерфейса используется библиотека Avalonia.

### **2 Функциональное назначение**

#### **2.1 Назначение программы**

Программа предназначена для изучения ЭВМ PDP-11, в частности языка ассемблера и работы внешних устройств (далее, ВУ).

Программа реализует следующие функции:

- Ассемблирование программ, написанных на языке ассемблера PDP-11;
- Исполнение машинного кода PDP-11;
- Подключение ВУ, совместимых с PDP-11.

Программа предоставляет возможность исполнять машинный код пошагово (по одной команде) и автоматически. Поддерживаются точки останова.

Пользователь способен просматривать состояние симулятора: карту памяти, значения регистров, подключенные внешние устройства.

Для подключения ВУ необходим драйвер, для создания которого предоставляется SDK.

## **2.2 Сведения о функциональных ограничениях на применение**

Программа не предназначена для работы под управлением ОС без поддержки графического интерфейса.

## **3 Описание логической структуры**

### **3.1 Алгоритм программы**

Алгоритмы ассемблера приведены в приложении А.

Алгоритмы исполнителя приведены в приложении Б.

Алгоритмы графического интерфейса приведены в приложении В.

### **3.2 Используемые методы**

**Метод шаблонного проектирования** – позволяет облегчить работу проектировщиков и разработчиков ПО и ускорить процесс разработки.

**Метод объектно-ориентированного программирования** – Применение метода объектно-ориентированного программирования позволяет разрабатывать сложные и в тоже время качественные программы.

**Тестирование** – Использование данного метода в значительной мере способствует повышению качества и оперативности выполнения.

**Тестирование по документации** – Применение данной методики тестирования программного обеспечения позволит повысить качество проверки программного обеспечения на наличие ошибок и достижения поставленных целей.

**Тестирование «Черного ящика»** – Данная технология описывает порядок проведения тестирования "черного ящика", используемого для обеспечения комплекса мероприятий проверки качества информационной системы.

### **3.3 Структура программы с описанием функций составных частей и связи между ними**

Структурная схема программы приведена в приложении Г.

### 3.4 Связи программы с другими программами

Связи с другими программами нет.

## 4 Используемые технические средства

В состав технических средств должен входить IBM-совместимый персональный компьютер включающий в себя:

- процессор с тактовой частотой не менее 2 ГГц;
- оперативную память объемом не менее 1 ГБ;
- свободное место на диске не менее 200 МБ.

## 5 Вызов и загрузка

Загрузка программы осуществляется набором в командной строке или терминале имени загрузочного файла: “*pdp11simulator*”.

## 6 Входные данные

Входными данными для графического интерфейса являются файл *appsettings.json*, расположенный рядом с исполняемым файлом и содержащий настройки шрифта окна редактора, файл проекта с расширением *pdp11proj*, имеющий структуру, приведенную в листинге 1, и файлы с исходными кодами на языке ассемблера. Также входными данными являются данные, вводимые пользователем,

Входными данными для ассемблера является файл проекта с расширением *pdp11proj* и файлы с исходными кодами на языке ассемблера.

Входными данными для исполнителя являются файл проекта и объектный файл с расширением *pdp11bin*, имеющий структуру, приведенную в листинге 2, где *command* – исходный код на языке ассемблера, апостроф – метка перемещаемого адреса.

Листинг 1. Структура файла проекта

```
{
  "Executable": "Путь к исполняемому файлу",
  "Files": ["Список файлов проекта"],
  "Devices": ["Список подключенных внешних устройств"],
  "StackAddress": "Начальный адрес указателя стека",
  "ProgramAddress": "Начальный адрес исполняемой программы в памяти"
```

```
}
```

Листинг 2. Структура объектного файла

```
000000;command  
000000'  
000000  
...
```

## 7 Выходные данные

Выходными данными графического интерфейса являются файл проекта, файлы с исходным кодом и файл *appsettings.json* (в случае изменения пользователем настроек шрифта).

Выходными данными ассемблера является объектный файл.

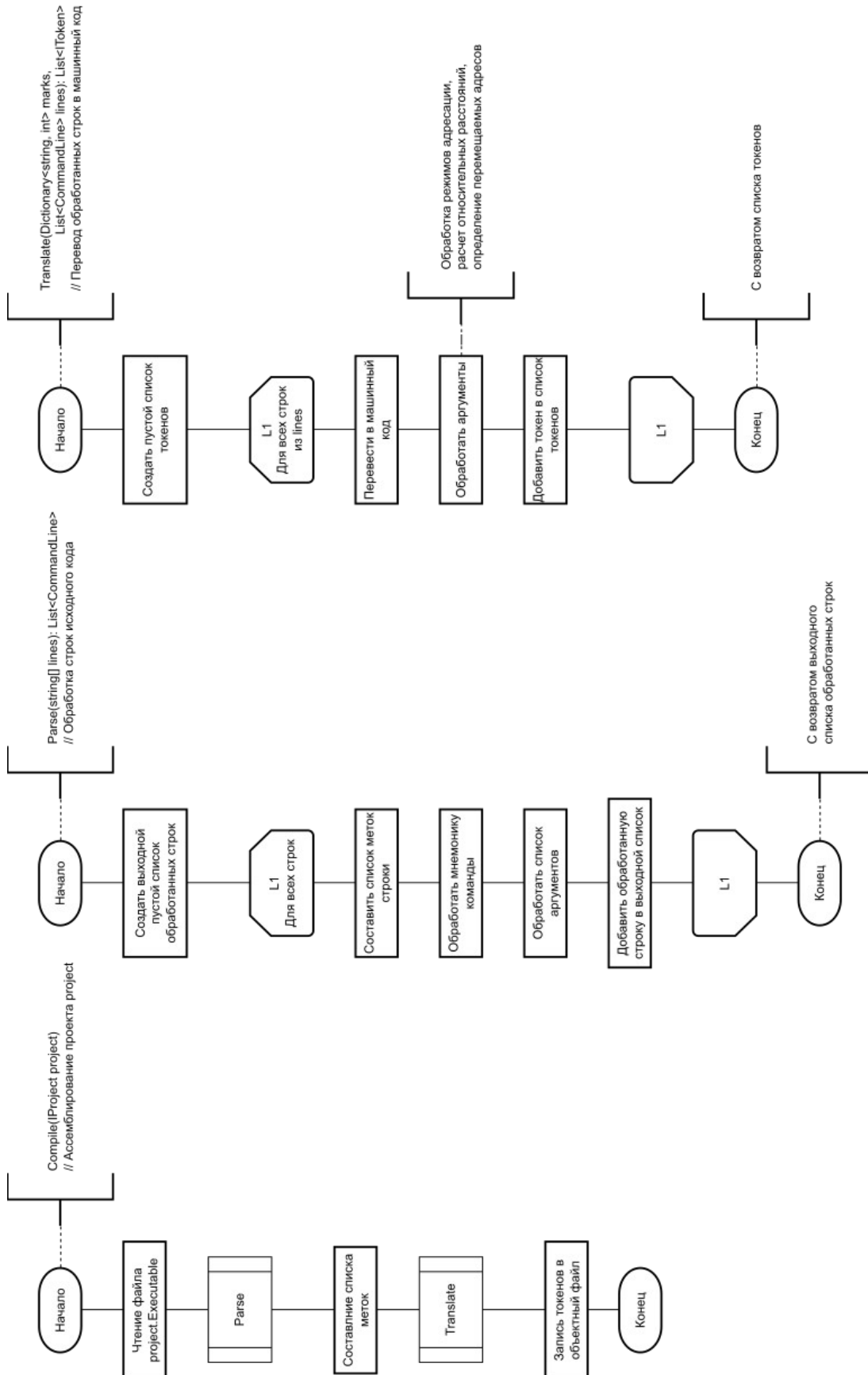
Также выходными данными являются различные сообщения пользователю.

<b>Лист регистрации изменений</b>
-----------------------------------

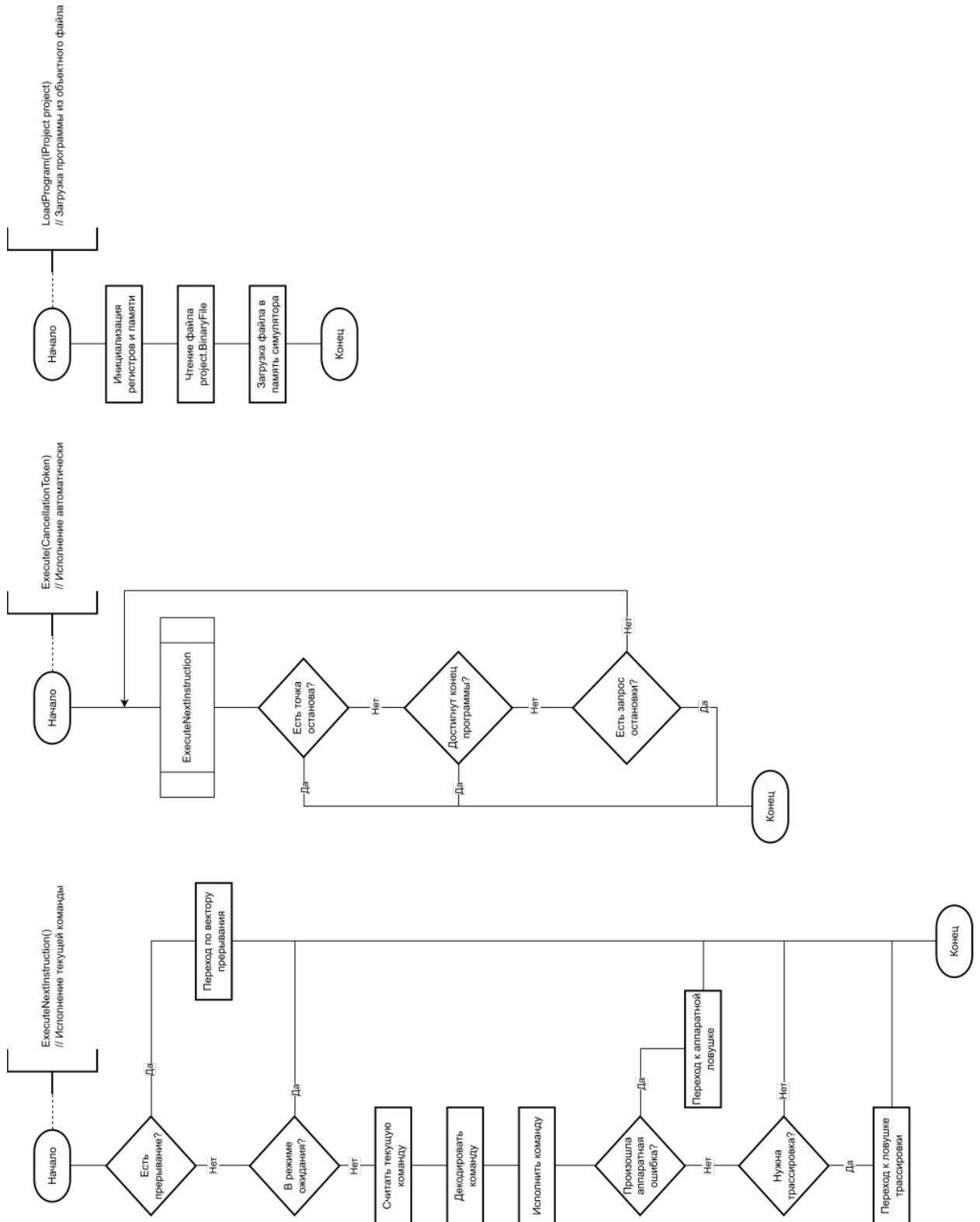
[illegible]



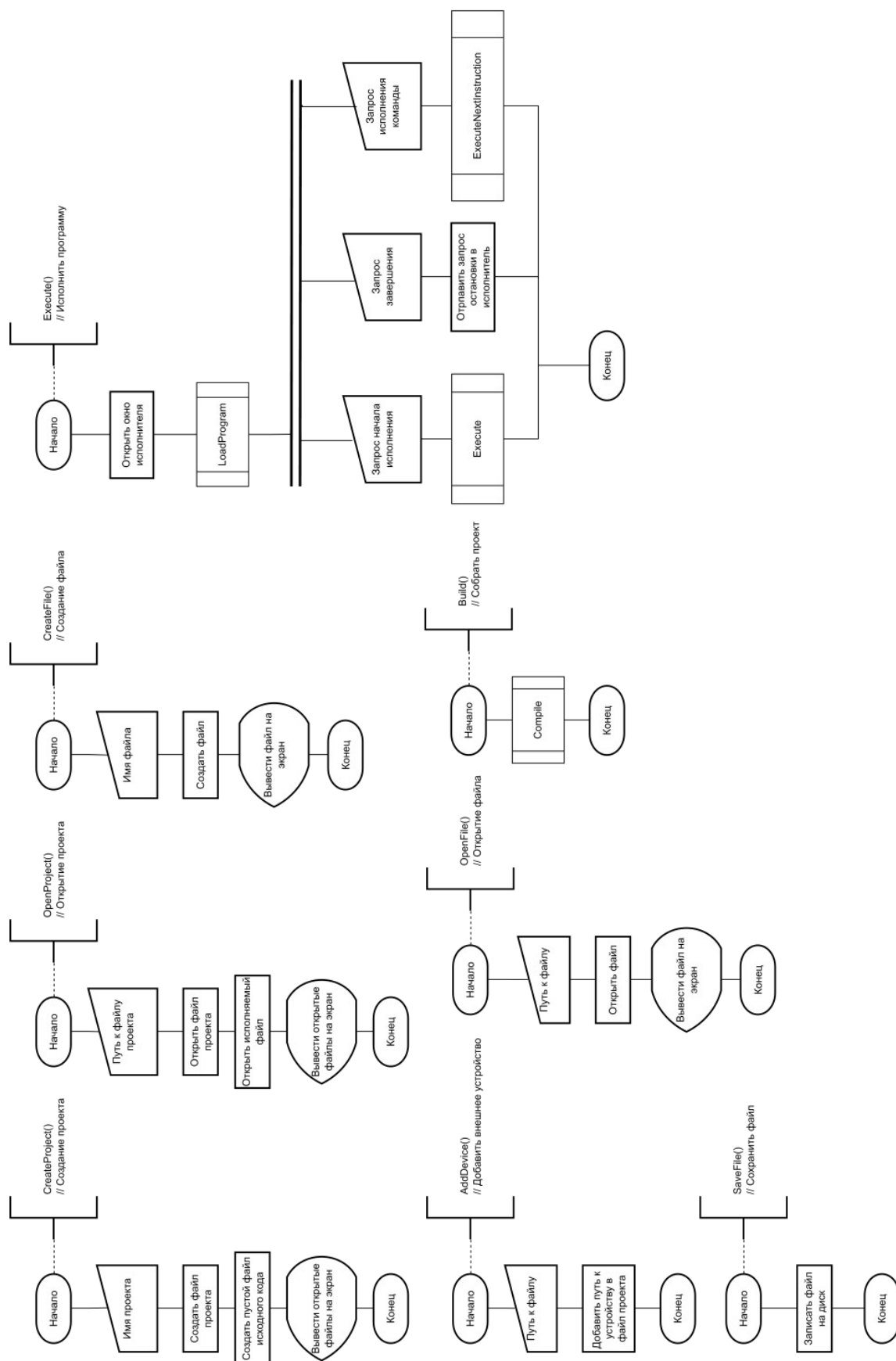
## Приложение А. Блок-схемы алгоритмов Ассемблера



## Приложение Б. Блок-схемы алгоритмов Исполнителя



## Приложение В. Блок-схемы алгоритмов Графического интерфейса



## Приложение Г. Структурная схема программы

