**Доклад**

**Слайд 2.**

На этом слайде представлена диаграмма прецедентов. На диаграмме показаны все возможные функциональные отношения.

**Слайд 3.**

При выборе инструментов я исходил из некоторых критериев.

Важность критерия я выбирал из: низкая, ниже средней, средняя, ниже высокой, высокая.

Исходя из этих критериев, я сравнил 3 языка программирования от 0 до 10 баллов за критерий.

По результатам сравнения был выбран язык программирования Haskell.

**Слайд 4.**

В данном разделе приведен сценарий использования программы пользователем.

Пользователь вводит необязательные аргументы программы, например ширину и/или высоту конечного изображения. Дальше вводится имя файла и формат для вывода изображения и имя файла с исходным кодом. После этого программа проверяет существование указанного файла, если файл не существует, выводится ошибка. Если файл существует то программа выводит блок-схему в указанный файл, в выбранном формате.

**Слайд 5.**

Здесь представлены все типы данных описанные в проекте.

**Слайд 6.**

В главный модуль подключаются 3 основных модуля: модуль с парсером, модуль с типом данных блок-схемы, и модуль с отрисовкой блок-схемы.

Также туда подключается вспомогательный модуль с универсальными функциями не зависящими от остальных модулей программы.

Далее в главном модуле считывается файл, производится его парсинг, и в случае успешного парсинга, с помощью функции readyProgramm АСД подготавливается для построение блок-схемы, затем строится блок-схема, и наконец отрисовывается в выходной файл.

Главный модуль является связующим звеном между главными модулями программы и позволяет инкапсулировать в себе всю работу с пользователем.

**Слайд 7.**

Разработаны 6 модулей, не включая главный:

* Функции парсера
* Типы данных АСД
* Вспомогательные функции для парсинга отступов
* Типы данных блок-схемы и функции для построения блок-схемы
* Функции для отрисовки блок-схемы
* Вспомогательные функции, не зависящие от остальных модулей