



ООО «А-Я эксперт»

# **Библиотека QSimHs для симуляции квантовых вычислений**

## **Руководство по установке**

Версия 1.0  
(Май 2023 г.)

Москва, 2023

# 1. Введение

Описываемое в настоящем документе программное обеспечение представляет собой библиотеку симуляции квантовых вычислений под названием QSimHs. Цель настоящего документа с инструкциями по установке — предоставить пошаговое руководство по получению и установке копии программного обеспечения, включая необходимые предварительные условия и системные требования. Документ также включает советы по устранению неполадок, чтобы помочь пользователям преодолеть любые проблемы, которые могут возникнуть во время установки.

Настоящий документ структурирован следующим образом:

- Введение: содержит краткий обзор назначения программного обеспечения и этого документа.
- Системные требования: перечисляет минимальные и рекомендуемые требования к аппаратному и программному обеспечению для установки и использования описываемой библиотеки.
- Предварительные условия: подробно описывает программное обеспечение и библиотеки, которые должны быть установлены перед установкой данного программного обеспечения.
- Получение и развёртывание библиотеки: предоставляет пошаговые инструкции по загрузке, установке и развёртыванию программного обеспечения библиотеки QSimHs.
- Устранение неполадок: описывает шаги, которые необходимо предпринять, если в процессе установки или развёртывания возникли проблемы.
- Заключение: завершает документ, резюмируя основные моменты и направляя пользователей к дополнительным ресурсам для получения дальнейшей помощи.

## 2. Системные требования

Ниже приведены минимальные системные требования для установки и запуска программного обеспечения библиотеки QSimHs:

- Операционная система: Windows 7 или более поздней версии, macOS 10.12 или более поздней версии, или последняя версия популярного дистрибутива Linux, либо любая иная операционная система, на которой может быть развёрнуто и запущено программное обеспечение Haskell Platform.
- Процессор: Intel Core i5 или выше, либо любой современный процессор с поддержкой набора инструкций SSE2.
- Оперативная память: 8 ГБ или выше.
- Дисковое пространство: 500 Мб свободного дискового пространства для установки программного обеспечения Haskell Platform и дополнительное пространство для любых данных или временных файлов, которые будут созданы программным обеспечением.
- Интернет-соединение: для загрузки и установки программного обеспечения и всех необходимых зависимостей требуется надёжное и стабильное подключение к Интернету.

Поскольку программное обеспечение библиотеки QSimHs написано на языке программирования Haskell, оно должно быть совместимо с любой операционной системой, поддерживающей программное обеспечение Haskell Platform. Однако важно отметить, что некоторые из зависимостей, требуемых программным обеспечением библиотеки QSimHs, могут иметь специфические для конкретной платформы требования к установке. Поэтому рекомендуется тщательно следовать инструкциям по установке, приведённым в документации на соответствующие библиотеки и внешние модули из числа зависимостей,

чтобы обеспечить успешную установку и работу программного обеспечения библиотеки QSimHs.

Программное обеспечение библиотеки QSimHs, описанное в настоящей документации, требует от пользователя знакомства с языком программирования Haskell и принципами квантовых вычислений. Предполагается, что пользователи должны обладать рабочими знаниями языка Haskell и понимать основы квантовых вычислений, включая квантовые гейты, квантовые схемы и квантовые алгоритмы. Рекомендуется, чтобы пользователи имели опыт работы с функциональным программированием и математическими концепциями, такими как линейная алгебра и комплексные числа.

### 3. Предварительные условия

Ниже перечислены необходимые условия для установки программного обеспечения библиотеки QSimHs.

- **Haskell Platform** версии 8.0.2 или выше. Haskell — это функциональный язык программирования, и программное обеспечение библиотеки QSimHs разработано с его использованием. Платформу Haskell Platform можно загрузить с официального сайта по адресу <https://www.haskell.org/platform/>.
- **Quipper** версии 0.9.0 или выше. Quipper — это проблемно-ориентированный язык программирования для реализации квантовых схем, и программное обеспечение библиотеки QSimHs частично использует его для генерации некоторых квантовых схем. Инструментаций для работы с языком Quipper можно загрузить с официального сайта Quipper по адресу <http://www.mathstat.dal.ca/~selinger/quipper/>.
- **Cabal** версии 2.0 или выше. Cabal — это менеджер пакетов для языка Haskell, который используется для установки зависимостей программного обеспечения. Система Cabal может быть установлен вместе с платформой Haskell Platform.

Библиотека QSimHs использует следующие модули языка Haskell, которые должны быть установлены в составе Haskell Platform перед началом работы:

- **Control.Arrow** — модуль языка Haskell, предоставляющий класс типов `Arrow` и связанные с ним функции для вычислений, осуществляемых с направленными ациклическими графами;
- **Control.Monad** — модуль языка Haskell, предоставляющий класс типов `Monad` и связанные с ним функции для организации последовательных вычислений, обработки ошибок и побочных эффектов;
- **Data.Bits** — модуль языка Haskell, предоставляющий побитовые операции и операции сдвига для целочисленных типов;
- **Data.Complex** — модуль языка Haskell, предоставляющий комплексные числа, включая арифметические, тригонометрические и гиперболические функции для их обработки;
- **Data.Char** — модуль языка Haskell, предоставляющий функции для манипулирования символами и предикаты классификации символов;
- **Data.Function** — модуль языка Haskell, предоставляющий функции высшего порядка для манипулирования другими функциями, такие как композиция, применение и каррирование;
- **Data.List** — модуль языка Haskell, предоставляющий функции для работы со списками, такие как `map`, `filter`, `foldr` и т. д.;

- **Data.Maybe** — модуль языка Haskell, предоставляющий тип `Maybe` и связанные с ним функции для работы с необязательными значениями;
- **Data.Numbers.Primes** — модуль языка Haskell, предоставляющий функции для генерации и факторизации простых чисел;
- **Data.Ord** — модуль языка Haskell, предоставляющий класс типов `Ord` и связанные с ним функции для упорядочивания и сравнения значений;
- **Data.Ratio** — модуль языка Haskell, предоставляющий рациональные числа, включая арифметические функции и функции преобразования таких чисел;
- **Data.Tuple** — модуль языка Haskell, предоставляющий тип `Tuple` и связанные с ним функции для работы с парами и другими кортежами;
- **GHC.Real** — модуль языка Haskell, предоставляющий классы типов `Real` и `Integral` и соответствующие функции для деления и преобразования целых чисел;
- **Quipper** — библиотека платформы Haskell Platform для описания и моделирования квантовых схем, включая одноимённый проблемно-ориентированный язык квантового программирования;
- **System.Random** — модуль языка Haskell, предоставляющий функции для генерации случайных чисел.

Пользователь должен убедиться, что эти предварительные условия выполнены, перечисленные в этом разделе программное обеспечение установлено и работает должным образом, прежде чем устанавливать программное обеспечение библиотеки QSimHs.

## 4. Получение и развёртывание библиотеки

В этом разделе содержатся подробные инструкции по загрузке и развёртыванию библиотеки QSimHs. Этот раздел охватывает два основных этапа: загрузку библиотеки QSimHs и её развёртывание в отдельную папку в операционной системе пользователя. Эти шаги необходимы для успешной установки библиотеки QSimHs.

1. **Загрузка библиотеки QSimHs.** Первым шагом является загрузка библиотеки QSimHs. Для этого необходимо перейти в официальный репозиторий QSimHs на GitHub по адресу <https://github.com/Roman-Dushkin/QSimHs>. Оттуда можно загрузить исходный код в виде zip-файла или клонировать репозиторий с помощью инструментов Git.
2. **Развёртывание библиотеки QSimHs.** После загрузки библиотеки QSimHs необходимо развернуть её в системе пользователя. Для этого необходимо создать новую папку в операционной системе и скопировать исходный код QSimHs в эту папку.
3. **Проверка установки.** Чтобы убедиться, что библиотека QSimHs была установлена правильно, можно попробовать импортировать её в произвольный модуль на языке Haskell. Для этого следует открыть интерпретатор GHCi, выполнив команду `ghci` в терминале. Находясь в интерпретаторе, следует ввести следующую команду:

```
Prelude> import Qubit
```

Если библиотека импортируется без ошибок, установка прошла успешно. Однако, если возникли какие-либо ошибки, следует обратиться к разделу 5 настоящего руководства, чтобы узнать, как решить проблему.

Важно отметить, что для использования библиотеки QSimHs пользователи должны обладать рабочими знаниями языка Haskell и модели квантовых вычислений. Кроме того,

могут существовать программные зависимости, которые необходимо установить до установки библиотеки QSimHs, как описано в разделе 3 настоящего руководства.

## 5. Устранение неполадок

Если пользователь столкнулся с какими-либо ошибками или проблемами при компиляции и запуске программ на языке Haskell или на языке Quipper, есть несколько шагов, которые можно предпринять для устранения неполадок.

Во-первых, если при компиляции кода компилятор сообщает, что отсутствуют какие-либо библиотеки, необходимо установить их с помощью утилиты `cabal`, входящей в состав Haskell Platform. Например, если при компиляции кода появляется сообщение об ошибке типа «Could not find module `Data.Map`», можно использовать следующую команду в терминале для установки недостающего пакета:

```
cabal install containers
```

Эта команда устанавливает пакет `containers`, который включает модуль `Data.Map`.

Если у пользователя нет интерпретатора проблемно-ориентированного языка Quipper, необходимо загрузить и установить его. Это можно сделать с помощью утилиты `cabal` или загрузив его с официального сайта языка. После установки интерпретатора Quipper можно запускать свой код на языке Quipper, вызывая интерпретатор в терминале:

```
quipper your-program.qp
```

Если у пользователя всё ещё возникают проблемы, можно обратиться за дальнейшей помощью к документации на язык Quipper или на форумы сообщества.

## 6. Заключение

В заключение следует отметить, что установка библиотеки для симуляции квантовых вычислений QSimHs — это относительно простой процесс. Следуя инструкциям, изложенным в этом руководстве, пользователи смогут установить библиотеку и начать использовать её для моделирования квантовых вычислений.

Сначала необходимо убедиться, что система пользователя соответствует минимальным системным требованиям, и что у пользователя установлены необходимые предварительные условия. Затем необходимо загрузить и развернуть библиотеку QSimHs в отдельную папку в пользовательской операционной системе. Наконец, необходимо проверить правильность установки библиотеки, импортировав её главный модуль в произвольный модуль на языке Haskell и запустив несколько базовых симуляций.

Если в процессе установки или развёртывания возникли какие-либо проблемы, следует обратиться к разделу 5 настоящего руководства.

Дополнительные ресурсы и документацию можно найти на официальном сайте платформы Haskell Platform, а также в книге Романа Душкина «Квантовые вычисления и функциональное программирование»:

**Душкин Р. В.** *Квантовые вычисления и функциональное программирование*. — М.: ДМК-Пресс, 2014. — 318 с., ил.

Кроме того, на официальном YouTube-канале Р. В. Душкина «Душкин объяснит» имеются плейлисты, охватывающие основные темы, использованные в библиотеке QSimHs:

- Квантовые технологии: <https://clck.ru/34HwBK>
- Функциональное программирование: <https://clck.ru/34HwBx>
- Линейная алгебра: <https://clck.ru/34HwAj>

ООО «А-Я эксперт» надеется, что библиотека QSimHs станет для всех полезным инструментом в исследованиях и экспериментах в области квантовых вычислений. Спасибо, что выбрали это программное обеспечение, и мы будем рады любым вашим отзывам и предложениям, присылаемым на адрес электронной почты [info@aia.expert](mailto:info@aia.expert).