**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ "МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ (НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)"**

**ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ**

**Кафедра 806 «Вычислительная математика и программирование»**

КУРСОВАЯ РАБОТА

по дисциплине: «*Фундаментальная информатика*»

на тему:

«*Машина Тьюринга*»

Выполнил:

студент 1 курса группы М8О-108Б-23

Хулагов Магомед-Амин Саидович

**Москва 2023**

**СОДЕРЖАНИЕ**

ВВЕДЕНИЕ

1. БИОГРАФИЯ АЛАНА ТЬЮРИНГА

2. ИСТОРИЯ СОЗДАНИЯ МАШИНЫ ТЬЮРИНГА

3. КРИТИКА МАШИНЫ ТЬЮРИНГА

4. ИДЕЯ, МЕТОД, АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ

5. СЦЕНАРИЙ ВЫПОЛНЕНИЯ РАБОТЫ. ТЕСТЫ

6. РАСПЕЧАТКА ПРОТОКОЛА

7. ДНЕВНИК ОТЛАДКИ

8. ВЫВОДЫ

**ВВЕДЕНИЕ**

Тема исследования, связанная с Машиной Тьюринга, остается актуальной в настоящее время. Эта концепция является фундаментальной для развития компьютерных наук, алгоритмов, теории вычислений и искусственного интеллекта. Изучение Машины Тьюринга продолжает вносить значительный вклад в наше понимание компьютерных систем, а также служит источником вдохновения для ученых и инженеров по всему миру.

**Объект исследования** – Машина Тьюринга.

**Целью курсовой работы** – изучить и понять принцип работы машины Тьюринга.

**Задание:** обмен местами разрядов двоичного числа, находящихся на чётных и нечётных позициях

**Оборудование:**

Процессор **12th Gen Intel® Core™ i5-12450H** с ОП **8 ГБ 3200 MHz DDR4**, НМД \_\_\_\_\_\_\_\_\_\_ Мб. Монитор **Встроенный монитор 15,6-дюймовый (1920 × 1080)**

**Программное обеспечение:**

Операционная система семейства **Linux**, наименование **Ubuntu** версия **22.04.3 LTS** интерпретатор команд bash версия 5.1.16.

Система программирования нет

Редактор текстов нет

Утилиты операционной системы pwd, who, whoami, ls, cd, mkdir, cp, mv, rm, rmdir, chmod, cat, touch, ps, grep, sort.

Местонахождение и имена файлов программ и данных на домашнем компьютере ~/lab5

**БИОГРАФИЯ АЛАНА ТЬЮРИНГА**

Алан Матисон Тьюринг родился 23 июня 1912 года в Мейдстоуне, графство Кент, Англия. Он был вторым ребенком в семье. Его отец был сотрудником Индийской цивильной службы, а мать занималась активизмом в женском движении.

С детства Тьюринг проявлял необычайные математические способности. Он был учеником школы Шерборна, где проявил талант в математике и науках. В 1931 году он поступил в Королевский колледж в Кембридже, где изучал математику.

Во время учебы в Кембридже Тьюринг занимался исследованиями в области математики и логики. Он стал заинтересован в проблеме принятия решений и вычислительных процессах, что впоследствии привело к разработке Машины Тьюринга.

В 1936 году Тьюринг опубликовал статью "Вычисление с неограниченными ресурсами", в которой предложил концепцию Машины Тьюринга. Он показал, что с помощью этой абстрактной машины можно эмулировать работу любого возможного компьютера и решать широкий класс проблем.

В начале Второй мировой войны Тьюринг присоединился к кодово-дешифровальной единице английской разведки. Он работал в Блетчли Парк в команде, которая разрабатывала методы для расшифровки кодов, используемых немецкой шифровальной машиной "Энигма". Тьюринг внес значительный вклад в успешное дешифрование кодов "Энигмы" и играл ключевую роль в развитии электромеханической машины, известной как "Бомба", для ускорения процесса расшифровки.

После войны Тьюринг продолжил исследования в области компьютерных наук. Он работал в Национальном физическом лаборатории и занимался разработкой и проектированием электронных компьютеров.

Однако жизнь Тьюринга оборвалась в 1952 году, когда он был арестован за гомосексуальные отношения, которые в то время были незаконными в Великобритании. Тьюринг был приговорен к химической кастрации, поскольку выбрал химическую методу вместо тюремного заключения. Это принесло серьезное травматическое воздействие на его физическое и психическое здоровье. В итоге, 7 июня 1954 года, Тьюринг совершил самоубийство.

Посмертно Алан Тьюринг был признан одним из величайших ученых XX века. Его вклад в развитие компьютерных наук, теории вычислений и искусственного интеллекта является фундаментальным. В 2013 году британское правительство официально извинилось перед Тьюрингом за его обращение, и его предоставлено посмертное помилование за его уголовные судимости.

Биография Алана Тьюринга служит напоминанием о его невероятных достижениях в науке и его важности для развития современной технологии.

**ИСТОРИЯ СОЗДАНИЯ МАШИНЫ ТЬЮРИНГА**

Алан Тьюринг, английский математик и логик, предложил идею Машины Тьюринга в своей статье в 1936 году. В то время вопрос "Entscheidungsproblem" (проблема принятия решений) был актуален в математике. Проблема заключалась в том, можно ли разработать алгоритм, который для любого математического утверждения указывает, верно оно или ложно.

Тьюринг попытался ответить на этот вопрос, предложив концепцию универсальной вычислительной машины. Он хотел показать, что некоторые математические проблемы являются неразрешимыми, то есть не могут быть решены с помощью алгоритма. Для этого ему понадобилась формальная модель, которая могла бы эмулировать работу любого другого вычислительного устройства и выполнять произвольные вычисления.

Идея Машины Тьюринга базировалась на использовании бесконечной ленты, разделенной на ячейки, где можно записывать и считывать символы. Машина состояла из головки, которая перемещалась по ленте, и конечного множества состояний. Головка могла выполнять операции чтения, записи и перемещения по ленте в зависимости от текущего состояния и символа в текущей ячейке. Переходы между состояниями определяли правила перехода, которые задавали функциональность машины.

Ключевое открытие Алана Тьюринга заключается в том, что с помощью всего лишь одной универсальной машины Тьюринга можно имитировать работу любого другого устройства Тьюринга. Это означает, что все вычислимые функции могут быть вычислены с помощью универсальной машины Тьюринга. Данное открытие позволило Тьюрингу сформулировать тезис о вычислимости, который утверждает, что любое алгоритмически разрешимое задание может быть выполнено на универсальной машине Тьюринга.

Универсальная машина Тьюринга стала основой для развития теории вычислений и компьютерных наук. Она помогла выделить фундаментальные понятия, такие как алгоритмы, вычислимость и вычислительная сложность. Кроме того, она послужила теоретическим обоснованием для создания современных компьютеров и программирования.

С помощью Машины Тьюринга можно было эмулировать работу любого другого вычислительного устройства. Тьюринг показал, что Машина Тьюринга может выполнять операции, которые соответствуют базовым вычислениям, таким как сложение, вычитание и т.д. Он также доказал, что существуют проблемы, которые не могут быть решены с помощью Машины Тьюринга, и это привело к понятию "неразрешимости".

Да, важным результатом Алана Тьюринга было доказательство существования так называемых "неразрешимых проблем". Тьюринг показал, что существуют определенные вычислительные задачи, для которых не существует алгоритма, способного дать окончательный ответ, будь то "да" или "нет". Этот результат стал одним из ключевых в теории вычислимости.

Одна из самых известных проблем, для которой Тьюринг доказал неразрешимость, - это "Проблема остановки" или "Проблема останавливается ли программа на данном входе?". Суть проблемы заключается в том, что невозможно написать общий алгоритм, который бы мог определить, остановится ли произвольная программа на данном входе или продолжит работу бесконечно.

Также известной проблемой, для которой Тьюринг показал неразрешимость, является "Проблема остановки для Машины Тьюринга". Проблема остановки для Машины Тьюринга является одной из важных проблем в теории вычислимости. Она возникает в контексте попытки определить, можно ли написать программу или алгоритм, который будет определять, остановится ли данная Машина Тьюринга для заданного входа.

Формально проблему остановки можно сформулировать следующим образом: дана Машина Тьюринга M и входные данные w, требуется определить, остановится ли M при обработке входа w или будет работать бесконечно долго.

Проблема остановки оказалась неразрешимой для общего случая. Это значит, что не существует алгоритма, который бы мог гарантированно определить, остановится ли любая Машина Тьюринга для любого входа. То есть не существует универсального алгоритма, который бы мог решить проблему остановки для всех Машин Тьюринга.

Доказательство неразрешимости проблемы остановки основано на методе диагонализации, предложенном Куртом Гёделем в 1931 году. Он использовал аналогичные идеи для доказательства неразрешимости проблемы остановки в формальных системах.

Неразрешимость проблемы остановки имеет важные последствия для теории вычислений. Она показывает, что существуют ограничения на возможности алгоритмического решения некоторых задач. Более общими понятиями, основанными на проблеме остановки, являются рекурсивное перечислимость и разрешимость. Эти концепции помогают классифицировать задачи по степени их вычислимости.

Важно отметить, что неразрешимость проблемы остановки относится к общему случаю Машин Тьюринга. Для конкретных или специальных классов Машин Тьюринга можно найти решение проблемы остановки, но для общего класса Машин Тьюринга невозможно написать универсальный алгоритм, который всегда определит остановку.

Эти примеры демонстрируют, что существуют некоторые вычислительные задачи, для которых нет единого алгоритма, способного верно ответить на них во всех случаях. Это означает, что не все проблемы могут быть решены с помощью Машины Тьюринга или любого другого формального вычислительного устройства. Такие проблемы называются "неразрешимыми".

Это важное открытие Алана Тьюринга имело глубокое влияние на развитие компьютерных наук и теории вычислений. Оно помогло понять границы возможностей вычислительных систем и построить теоретический фундамент для изучения сложности алгоритмов и разработки эффективных вычислительных методов.

Идеи Тьюринга оказали огромное влияние на развитие компьютерной науки и информатики. Они стали основой для теории вычислений и помогли формализовать понятия алгоритма, вычислимости, сложности и многих других ключевых понятий в компьютерных науках. Машина Тьюринга стала важным шагом в развитии реальных компьютеров и систем, которые исполняют алгоритмы и решают сложные вычислительные задачи.

Таким образом, Машина Тьюринга, предложенная Аланом Тьюрингом, является одним из наиболее важных и фундаментальных достижений в истории компьютерных наук. Ее концепции и понятия оказали большое влияние на развитие вычислительной техники, алгоритмов и теории вычислений.

**КРИТИКА МАШИНЫ ТЬЮРИНГА**

Машина Тьюринга обладает фундаментальными ограничениями, которые вызывают некоторую критику. Вот некоторые из наиболее распространенных аргументов против машины Тьюринга:

* Абстрактность идеальной машины: Машина Тьюринга является абстрактной моделью, которая не учитывает реальные ограничения физического мира, такие как время и память. Она предполагает неограниченные ресурсы и бесконечную ленту, что не соответствует реальным вычислительным системам. Критики утверждают, что эта абстракция может приводить к непрактичным результатам при анализе сложности реальных задач.
* Отсутствие параллелизма: Машина Тьюринга основывается на последовательной обработке данных головкой, что не учитывает преимущества параллельных вычислений. В реальных компьютерных системах параллелизм широко используется для ускорения вычислений, и поэтому критики считают, что модель Тьюринга не отражает полностью реальный мир вычислений.
* Недостаток интуитивного понимания: Некоторые аспекты машины Тьюринга, такие как бесконечная лента и конечное число состояний, могут быть сложными для интуитивного понимания. Понятия, такие как эффективность алгоритмов и сложность задач, часто требуют более простых и наглядных подходов для понимания и объяснения.
* Необходимость абстрактных понятий: Для работы с машиной Тьюринга требуется понимание и использование абстрактных концепций, таких как состояния и символы. Это может оказаться сложным для новичков и неспециалистов, что затрудняет использование модели для практических целей.
* Проблема определения формальности: Некоторые критики утверждают, что машина Тьюринга определяет формальные алгоритмы, но не предоставляет четкого определения того, что такое формальность. В результате возникают различные интерпретации и неоднозначности, что усложняет применение модели в практической работе.

Не смотря на эти критические замечания, машина Тьюринга остается фундаментальной моделью вычислений и инструментом для анализа сложности алгоритмов и исследования теоретических вопросов вычислимости. Она имеет широкое применение в теоретической информатике и продолжает обеспечивать основу для развития компьютерных наук.

**ИДЕЯ, МЕТОД, АЛГОРИТМ, РЕШЕНИЯ ЗАДАЧИ**

Для начала опишем, как в общих чертах работает машина Тьюринга.

Машина Тьюринга - это простая вычислительная модель, придуманная английским математиком Аланом Тьюрингом. Она состоит из бесконечной ленты, разделенной на ячейки, головки, которая может двигаться по ленте и читать и записывать символы, и программы - набора инструкций, которые говорят головке, что делать в каждой конкретной ситуации.

Машина Тьюринга использует символы для представления данных на ленте. В данном случае нам нужно работать с двоичными числами, поэтому мы будем использовать символы "0" и "1". Лента может быть пустой или содержать некоторое двоичное число.

Основная идея генерации чисел из четных и нечетных разрядов двоичного числа заключается в том, что мы будем двигать головку по ленте, читая символы и записывая их в соответствующие области памяти для хранения четных и нечетных разрядов числа.

Давайте распишем подробно шаги алгоритма для генерации двух чисел.

Шаг 1: Инициализация

Запишем двоичное число на ленту. Пусть это будет число "101010". Предположим, что мы начинаем с головки в начале ленты и лента будет содержать только это число. Остальные ячейки ленты будут пустыми.

Установим указатель на текущую позицию на головке в начальную позицию.

Шаг 2: Чтение и запись символов

Пока головка находится в пределах числа (т.е. пока головка находится в позиции, где есть символы "0" или "1"), продолжаем выполнять следующие шаги.

Считываем символ из текущей позиции на ленте.

Если символ - "0", записываем его в область памяти для хранения четных разрядов числа.

Если символ - "1", записываем его в область памяти для хранения нечетных разрядов числа.

Сдвигаем головку на следующую позицию вправо (если символ был "0") или две позиции вправо (если символ был "1").

Шаг 3: Вывод результатов

Дошли до конца числа на ленте.

Результатом генерации являются два числа: одно, составленное из четных разрядов, и другое - из нечетных разрядов.

Мы можем сохранить эти числа в отдельных областях памяти на ленте или вывести их на экран.

Шаг 4: Завершение

Программа завершается, машина Тьюринга останавливается.

Это подробная расшифровка шагов для генерации чисел из четных и нечетных разрядов двоичного числа на машине Тьюринга. При необходимости можно дополнить алгоритм дополнительными инструкциями или условиями в зависимости от конкретной задачи.

**ТЕСТЫ:**

Тесты:

Входное сообщение:

\_10101\_

Выходное сообщение:

\_10101\_111\_0

Входное сообщение:

\_101\_

Выходное сообщение:

\_101\_11\_0\_

Входное сообщение:

\_10\_

Выходное сообщение:

\_10\_1\_0\_

Входное сообщение:

\_1\_

Выходное сообщение:

\_1\_1\_

Входное сообщение:

\_00\_

Выходное сообщение:

\_00\_0\_0\_

Входное сообщение:

\_1000\_

Выходное сообщение:

\_1000\_1\_0

Входное сообщение:

\_1001\_

Выходное сообщение:

\_1001\_10\_1

Входное сообщение:

\_1011011\_

Выходное сообщение:

\_1011011\_1101\_11\_

Входное сообщение:

\_100\_

Выходное сообщение:

\_100\_10\_0

**РАСПЕЧАТКА ПРОТОКОЛА**

**00, ,<,01**

**01,0,<,01**

**01,1,<,01**

**01, ,>,02**

**02,1, ,03**

**03, ,>,04**

**04,1,>,04**

**04,0,>,04**

**04, ,>,05**

**05,1,>,05**

**05,0,>,05**

**05, ,1,06**

**06,1,<,06**

**06,0,<,06**

**06, ,<,07**

**07,1,<,07**

**07,0,<,07**

**07, ,1,08**

**08,1,>,10**

**10,1,>,02**

**10,0,>,02**

**10, , ,02**

**51, ,>,82**

**82,1,>,82**

**82,0,>,82**

**82, ,>,83**

**83,0, ,83**

**83, ,>,83**

**83,1,>,84**

**84, , ,84**

**84,0,>,84**

**84,1,>,84**

**42, ,>,85**

**85,1,>,85**

**85,0,>,85**

**85, ,>,86**

**86,0, ,86**

**86,1,>,88**

**88,1,>,88**

**88,0,>,88**

**88, , ,88**

**86, , ,87**

**87, , ,122**

**122, ,>,123**

**123,1,>,123**

**123,0,>,123**

**123, , ,124**

**124, ,<,124**

**124,0,>,125**

**125, , ,125**

**124,1,>,126**

**126, ,>,127**

**127, ,0,128**

**128,0,>,128**

**128, , ,128**

**02,0, ,20**

**20, ,>,21**

**21,1,>,21**

**21,0,>,21**

**21, ,>,22**

**22,1,>,22**

**22,0,>,22**

**22, ,0,23**

**23,0,<,23**

**23,1,<,23**

**23, ,<,24**

**24,1,<,24**

**24,0,<,24**

**24, ,0,25**

**25,0,>,26**

**26,0,>,02**

**26,1,>,02**

**26, , ,02**

**02, ,<,40**

**40,1,<,40**

**40,0,<,40**

**40, ,>,41**

**41,1,>,42**

**41,0,>,42**

**42,1, ,43**

**43, ,>,44**

**44,1,>,44**

**44,0,>,44**

**44, ,>,45**

**45,1,>,45**

**45,0,>,45**

**45, ,>,46**

**46,0,>,46**

**46, ,1,47**

**47,0,<,47**

**46,1,>,46**

**47,1,<,47**

**47, ,<,48**

**48,1,<,48**

**48,0,<,48**

**48, ,<,49**

**49,1,<,49**

**49,0,<,49**

**49, ,1,50**

**50,1,>,51**

**51,1,>,42**

**51,0,>,42**

**42,0, ,60**

**60, ,>,61**

**61,1,>,61**

**61,0,>,61**

**61, ,>,62**

**62,1,>,62**

**62,0,>,62**

**62, ,>,63**

**63, ,0,64**

**63,1,>,63**

**64,1,>,64**

**64,0,<,65**

**65,1,<,65**

**65,0,<,65**

**65, ,<,66**

**63,0,>,63**

**66,1,<,66**

**66,0,<,66**

**66, ,<,67**

**67,1,<,67**

**67,0,<,67**

**67, ,0,68**

**68,1,>,69**

**68,0,>,80**

**69,1,>,80**

**69,0,>,80**

**80,1,>,42**

**80,0,>,42**

**80, ,>,130**

**130,1,>,130**

**130,0,>,130**

**130, ,>,131**

**131,1,>,131**

**131,0,>,131**

**131, , ,131**

**ДНЕВНИК ОТЛАДКИ**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб.  Или  Дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
| 1 | Дом | 26.09.2023 | 13:48 | Отсутствие удаления лидирующих нулей. | Добавил проверку и последующее удаление первого элемента в скопированной части, после чего проверяю следующий элемент на подобное равенство. |  |

**ВЫВОД**

Машина Тьюринга представляет собой основу для понимания возможностей и ограничений алгоритмов и вычислений. Ее абстрактная структура позволяет легко понять концепции, связанные с алгоритмами и решением задач.

Как уже упоминалось, устройство состоит из бесконечной ленты, разделенной на ячейки, каждая из которых может содержать символ из некоторого предопределенного алфавита. Головка машины способна считывать символ из текущей ячейки ленты, записывать новый символ в эту ячейку и перемещаться влево или вправо на одну ячейку. Операции считывания, записи и перемещения головки являются основой для выполнения вычислительных операций.

Машина Тьюринга демонстрирует, что даже с простейшими правилами и ограниченным набором операций можно выполнять сложные вычисления. Это полезная абстракция, позволяющая изучать основные принципы теории вычислений.

Таким образом, машина Тьюринга является ключевым понятием в теории алгоритмов и предоставляет основу для понимания возможностей вычислительных устройств. Машина Тьюринга является основой для понимания того, что называется "вычислимостью". Идея Алана Тьюринга заключалась в том, чтобы создать простую абстрактную модель, способную моделировать работу любого алгоритма, независимо от его сложности. Эта модель, несмотря на свою простоту, демонстрирует, что с ее помощью можно эмулировать выполнение любого алгоритма, что делает ее универсальной в рамках теории вычислений.

Машина Тьюринга также играет важную роль в теории алгоритмов, поскольку она дает возможность формально понимать, что означает "вычислимость". В теории, если алгоритм можно представить как последовательность инструкций, которые можно выполнить на машине Тьюринга, то этот алгоритм считается "вычислимым".

Таким образом, понимание Машины Тьюринга играет важную роль не только в теоретической информатике, но также имеет практическое применение, так как многие принципы, которые лежат в ее основе, используются при разработке и анализе алгоритмов, программ и компьютерных систем.  
  
  
  
  
Машина Тьюринга основана на простом наборе правил, которые определяют ее поведение в зависимости от текущего состояния и символа, с которым головка взаимодействует. В основе этих правил лежит концепт конечного списка состояний, переходы между состояниями и изменение содержимого ленты.

Машина Тьюринга является вычислительно эквивалентной другим моделям вычислений, таким как рекурсивные функции или λ-исчисление, и доказала, что некоторые задачи неразрешимы. Например, проблема остановки (определение, остановится ли данная программа или машина Тьюринга при заданных входных данных) является неразрешимой для машин Тьюринга.

Машина Тьюринга является фундаментальным понятием в теории вычислимости и теоретической информатике. Она является основой для разработки алгоритмов и компьютерных программ. Машина Тьюринга и ее концепции широко используются в области алгоритмической сложности, формальных языков, компиляторов, искусственного интеллекта и других областях информатики.

Машина Тьюринга имеет свои ограничения, например, она не учитывает ограничение ресурсов, таких как время и память. Однако, она все равно является мощным инструментом для анализа сложности задач и исследования основных принципов вычислений.