**Министерство науки и высшего образования Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Тверской государственный технический университет» (ТвГТУ)**

**Кафедра информационных систем**

**КОНТРОЛЬНЫЕ ЗАДАНИЯ**

**ПО ДИСЦИПЛИНЕ: «ТЕОРИЯ АЛГОРИТМОВ»**

Выполнил: студент 2 курса

группы Б.ИСТ.РВС.35.20

факультета ИДПО

Всеволожский Владимир Никитович

студенческий билет № 20335

Проверил: д.т.н. Богатиков В.Н.

Тверь 2022

**Контрольная работа №1 ТА.**

**Задание 1.5 (Рекурсия)**

Определить функцию, полученную из функций и по схеме примитивной рекурсии:

*,*

Решение:

Схема примитивной рекурсии:

При

При ,

По схеме примитивной рекурсии найдем значения :

Можно предположить, что

Докажем последнюю формулу методом математической индукции по переменной y:

1. Проверим формулу при

, т. е. при формулы верна

1. Допустим, что предположение индукии верно при , то есть верна формула

Докажем, что предположение индукции верно при , т.е. верна формула . Выразим с помощью схемы примитивной рекурсии.

Таким образом, на основании метода математической индукции формула доказана для всех

Ответ:

**Задание 2.5**

По заданной машине Тьюринга T и начальной конфигурации найти заключительную конфигурацию:

Решение:

Таблица выполнения алгоритма будет иметь вид:

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 0 |  |  |
| 1 |  |  |
|  |  |  |

1. Схематически рассмотрим начальную конфигурацию на ленте машины Тьюринга:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | 1 | 0 | 0 | 1 |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

1. В дальнейшем проследим за движением каретки в соответствии с заданным алгоритмом T:

А) каретка из начальной конфигурации сменит собственное значение на и сдвинется на шаг влево по ленте (L).

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | 1 | 0 | 0 | 1 |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

Б) сменит значение на ленте на 1 и поменяет собственное значение на и каретка останется на месте (E) и алгоритм остановится, т.к. достигнуто значение .

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | 1 | 1 | 0 | 1 |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

1. Таким образом получаем заключительную конфигурацию:

Ответ:

**Задание 3.5 (Нормальные алгоритмы)**

Пусть и

Как действует данный алгоритм?

Применим ли данный алгоритм к любому слову в алфавите ?

Решение:

Рассмотрим данный нормальный алгоритм (систему подстановок) и выделим приоритет представленных правил подстановки:

1. правило преобразующее с в с, означающее, что ни чего меняться не будет
2. завершающая формула подстановки, терминальное правило
3. завершающая формула подстановки, терминальное правило

В нашем случае нормальный алгоритм не применим ни к одному из возможных слов, т.к. «с» заменится на «с», а с остальными буквами исходного алфавита произойдет завершение алгоритма.

**Контрольная работа №2 ТА.**

**Контрольное задание по сортировке массивов (вариант 3).**

Разработать программу вычислительного эксперимента для исследования характеристик эффективности сортировки прямого включения. Проверить совпадение модельных представлений и результатов вычислительного эксперимента.

Предусмотреть вывод исходной и результирующей информации в файл. Оценить сложность программы.

Решение:

Для решения задачи использовано программное обеспечение Microsoft Visual Studio 2022 работающее с условно компилируемым языком программирования C#.

В результате получился исходный код, который при компиляции будет выводить сортировку элементов массива методом сортировки прямого включения.

Пример функции реализующей метод сортировки прямого включения:

        private void directInclusionSort (double[] arr)

        {

            for (int i = 1; i < arr.Length; i++)

            {

                double tmp = arr[i];

                for (int j = i - 1; j >= 0 && arr[j] > tmp; j--)

                    arr[j + 1] = arr[j];

                arr[j + 1] = tmp;

            }

        }

Предусмотрен также вывод исходной и результирующей информации в один или разные файлы.