

Задача А. Чётность числа нулей

Имя выходного файла: `zero.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Это ознакомительная задача. В ней нужно посчитать чётность числа нулей, записанных на ленте и допустить строки с чётным числом нулей.

Например, это можно сделать с помощью такого кода:

```
start: s
accept: ac
reject: rj
blank: _
s _ -> ac _ ^
s 0 -> n _ >
n 0 -> s _ >
n _ -> rj _ >
```

В систему нужно посылать файл с соответствующим названием в условии. Символы и состояния — строки. Направления переходов — `<`, `>` или `^`. Изначально головка стоит в начале входа. Всё вне слова заполнено символом `blank`. Если попытаться пойти по правилу, которого нет в машине, вы автоматически попадёте в отвергающее состояние.

Формат входного файла

На ленте записано от одного до десяти нулей.

Формат выходного файла

Машина должна перейти в допускающее состояние, если число нулей чётно, и в отвергающее иначе.

Примеры

лента в начале	лента в конце
0	<i>rejected</i>
0 0	<i>accepted</i>

Задача В. Сложение двух чисел

Имя выходного файла: `aplusb.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Реализуйте сложение двух чисел на одноленточной машине Тьюринга.

Формат входного файла

На ленте через `+` записаны два числа a и b ($0 \leq a \leq 2^{15}$) в двоичной системе счисления без ведущих нулей.

Формат выходного файла

В конце на ленте должна быть записана сумма чисел в том же формате, головка должна указывать на начало этого числа, помимо этого числа на ленте ничего не должно быть. Слово должно быть допущено.

Примеры

лента в начале	лента в конце
1 1 + 1 0	<i>accepted</i> 1 0 1

Задача С. Зеркальное отображение

Имя выходного файла: `mirror.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

На ленте записано слово w . Требуется найти слово $s = ww'$, где w' - слово w , записанное в обратном порядке. Реализуйте алгоритм на одноленточной машине Тьюринга.

Формат входного файла

На ленте записано слово w ($w \in \{0,1\}^*$, $1 \leq |w| \leq 200$). Головка улавливает на первую букву слова.

Формат выходного файла

В конце на ленте должно быть записано слово s , головка должна указывать на начало этого слова. На ленте ничего не должно быть. Слово должно быть допущено.

Примеры

лента в начале	лента в конце
1 0 1 0 0	<i>accepted</i> 1 0 1 0 0 0 0 1 0 1

Задача D. Тандемный повтор

Имя выходного файла: `tandem.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Проверьте является ли слово тандемным повтором на одноленточной машине Тьюринга.

Формат входного файла

На ленте записано слово s ($1 \leq |s| \leq 100$), состоящее из символов 0 и 1.

Формат выходного файла

Машина Тьюринга должна завершаться в допускающем состоянии, если слово является тандемным повтором, и в отвергающем в противном случае.

Примеры

лента в начале	лента в конце
1 1 0 1 1 0	<i>accepted</i>

Задача Е. Сбалансированные скобки

Имя выходного файла: `balanced.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Реализуйте на машине Тьюринга проверку слова на принадлежность языку правильных скобочных последовательностей. Напомним, что правильной скобочной последовательностью называется строка, удовлетворяющая грамматике $S = \varepsilon \mid (S)S$.

Формат входного файла

На ленте написана последовательность открывающих и закрывающих скобок. Её длина не превышает 100 символов.

Формат выходного файла

После работы машина Тьюринга должна перейти в допускающее состояние, если исходное слово принадлежит языку правильных скобочных последовательностей и в отвергающее состояние иначе.

Примеры

лента в начале	лента в конце
	<i>accepted</i>
)	<i>rejected</i>
()	<i>accepted</i>

Задача F. Развернутое слово

Имя выходного файла: `reverse.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

На ленте записано слово w . Требуется найти слово $s = w'$, где w' - слово w , записанное в обратном порядке. Реализуйте алгоритм на одноленточной машине Тьюринга.

Формат входного файла

На ленте записано слово w ($w \in \{0,1\}^*$, $1 \leq |w| \leq 200$). Головка улавливает на первую букву слова.

Формат выходного файла

В конце на ленте должно быть записано слово s , головка должна указывать на начало этого слова. На ленте ничего не должно быть. Слово должно быть допущено.

Примеры

лента в начале	лента в конце
1 0 1 0 0	<i>accepted</i> 0 0 1 0 1

Задача G. Сравнение двух чисел

Имя выходного файла: `less.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Реализуйте оператор меньше для двух чисел на одноленточной машине Тьюринга.

Формат входного файла

На ленте через `<` записаны два числа a и b ($|a|, |b| \leq 1000$) в двоичной системе счисления без ведущих нулей.

Формат выходного файла

Машина должна перейти в допускающее состояние, если первое число меньше второго, и в отвергающее иначе.

Примеры

лента в начале	лента в конце
<code>1 1 < 1 0</code>	<i>rejected</i>
<code>0 < 1</code>	<i>accepted</i>

Задача Н. Из троичной в двоичную.

Имя выходного файла: `convertto2.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

На ленте записано число w в троичной системе счисления. Требуется перевести его в двоичную систему счисления. Реализуйте алгоритм на одноленточной машине Тьюринга.

Формат входного файла

На ленте записано число w без ведущих нулей. ($1 \leq |w| \leq 9$). Головка указывает на первую цифру числа.

Формат выходного файла

В конце на ленте должно быть записано число s , представляющее собой двоичную запись числа w . Головка должна указывать на первую цифру числа s . Помимо этого числа на ленте ничего не должно быть. Число должно быть допущено.

Примеры

лента в начале	лента в конце
1 0 2	<i>accepted</i> 1 0 1 1

Задача I. Умножение двух чисел

Имя выходного файла: `multiplication.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Реализуйте умножение двух чисел на одноленточной машине Тьюринга.

Формат входного файла

На ленте через `*` записаны два числа a и b ($10000 \geq a \geq b \geq 0, a*b \leq 10000$) в двоичной системе счисления без ведущих нулей. Биты записаны от старшего к младшему. То есть числу 6 соответствует двоичное число 110.

Формат выходного файла

В конце на ленте должно быть записано произведение чисел в том же формате, головка должна указывать на начало этого числа, помимо этого числа на ленте ничего не должно быть. Слово должно быть допущено.

Примеры

лента в начале	лента в конце
1 1 * 1 0	<i>accepted</i> 1 1 0

Задача J. Логические выражения в постфиксной записи

Имя выходного файла: postfixlogic.out
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Реализуйте вычисление логических выражений, состоящих из констант, конъюнкций и дизъюнкций на многоленточной машине Тьюринга.

Формат входного файла

На первой ленте записано логическое выражение в постфиксной (обратной польской) записи. Значения символов следующие:

- 0 — ложь
- 1 — истина
- o — логическое “или”
- a — логическое “и”

Длина выражения не превышает 200 символов.

Формат выходного файла

В конце работы головка первой ленты должна указывать на вычисленное значение логического выражения.

Примеры

лента в начале	лента в конце
1 0 a 1 o	<i>accepted</i> 1
0 1 o 0 o 1 a 0 a	<i>accepted</i> 0

Задача К. Логические выражения в инфиксной записи

Имя выходного файла: `infixlogic.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Реализуйте вычисление логических выражений, состоящих из констант, конъюнкций и дизъюнкций на многоленточной машине Тьюринга.

Формат входного файла

На первой ленте записано логическое выражение в инфиксной записи. Значения символов следующие:

- 0 — ложь
- 1 — истина
- o — логическое “или”
- a — логическое “и”

Кроме перечисленных символов на ленте могут присутствовать открывающие и закрывающие скобки ((и) соответственно). Логическое “и” имеет больший приоритет, чем логическое “или”. Длина выражения не превышает 200 символов.

Формат выходного файла

В конце работы головка первой ленты должна указывать на вычисленное значение логического выражения.

Примеры

лента в начале	лента в конце
1 a 0 o 1	<i>accepted</i> 1
(0 o 0 o 1) a 1 a 0	<i>accepted</i> 0

Задача L. Вычисление факториала

Имя выходного файла: `factorial.out`
Ограничение по времени: 10 000 000 шагов
Максимальное число состояний: 500

Реализуйте вычисление факториала двоичного числа на одноленточной машине Тьюринга

Формат входного файла

На ленте через записано одно число a ($1 \leq a \leq 30$) в двоичной системе счисления без ведущих нулей.

Формат выходного файла

В конце на ленте должен быть записан факториал числа a в том же формате, головка должна указывать на начало этого числа, помимо этого числа на ленте ничего не должно быть. Слово должно быть допущено.

Примеры

лента в начале	лента в конце
1 0 1	<i>accepted</i> 1 1 1 1 0 0 0

Задача М. Сортировка

Имя выходного файла: `sorting.out`
Ограничение по времени: 100 000 000 шагов
Максимальное число состояний: 500

Реализуйте сортировку набора чисел на многоленточной машине Тьюринга.

Формат входного файла

На первой ленте через символ «|» записаны n чисел ($1 \leq n \leq 20$) a_i в двоичной системе счисления без ведущих нулей, старшие биты — слева ($0 \leq a_i < 1024$).

Само число n на ленте не записано.

Формат выходного файла

В результате вычислений на первой ленте должны быть записаны те же n чисел в таком же формате, в порядке неубывания. Головка первой ленты должна указывать на начало первого числа; кроме чисел и разделителей, на ленте ничего не должно быть. Слово должно быть допущено.

Примеры

лента в начале	лента в конце
1 1 0 1 1 1 0 1	<i>accepted</i> 1 1 1 0 1 1 1 0