Санкт-Петербургский политехнический университет Петра Великого

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

**Отчёт по лабораторной работе № 1**

**Дисциплина:** Низкоуровневое программирование

**Тема:** Машина Тьюринга

Выполнил студент гр. 3530901/90003 Зотов М. С.

(подпись)

Преподаватель Алексюк А.О.

(подпись)

“ ” 2021 г.

Санкт-Петербург

2021

**Оглавление**

[1. ТЗ 3](#_Toc64333607)

[2. Метод решения 3](#_Toc64333608)

[3. Описание состояний 5](#_Toc64333609)

[4. Работа программы 7](#_Toc64333610)

# **1. ТЗ**

Определение максимального из двух чисел, представленных в двоичном коде.

# **2. Метод решения**

Требования к исходным данным:

1. Между двумя сравниваемыми числами находится как минимум один пробел
2. Ячейки, на которых размещены два сравниваемых числа и расположенные между этими числами пробел(ы), слева и справа соседствуют с символом x, то есть ячейки ленты должны иметь следующий вид: |x|a1|a2|…|an|пробел(ы)|b1|b2|…|bm|x|, где ai и bi –

i-ые разряды n-разрядного числа A и m-разрядного числа B соответственно

1. Допускается сравнение чисел разной разрядности и чисел с незначащими нолями, то есть запись вида x00101\_010x вполне допускается – в данном случае будет производится сравнение 101 и 10
2. В начальный момент времени головка должна быть размещена либо на символе x, который стоит слева, либо на старшем разряде левого числа

Результатом работы программы будет положение головки на нуле, если A максимальное, на единице, если B максимальное, на символе x, если A = B

Идея решения следующая:

Первым делом происходит избавление от незначащих нулей у обоих чисел – просто, начиная со старшего разряда левого числа A убираем все нули, пока не встретим единицу. Если мы прошли по всем разрядам числа A (то есть вышли к ячейке с пробелом) и ни разу не обнаружили 1, значит, A равно 0, и, так как все нули мы убирали, надо вернуться на одну ячейку влево и добавить туда нолик, иначе A будет просто пустым. Аналогично поступаем с B – так же удаляем все нули, пока не встретим в B единицу, если единицу не встретили (т.е. B = 0), так же добавляем в B один нолик. После избавления от незначащих нулей возвращаемся в левый x.

В результате получили n-разрядное число A и m-разрядное число B:

a1 a2 … an

b1 b2 … bm

Теперь с i = 1 начинаем сравнивать i-ые разряды, а после сравнения уменьшаем разрядность A и B путем удаления i-ых (то есть текущих старших) разрядов. Продолжаем так делать, пока не окажется, что ai != bi. Таким образом, у нас есть 2 варианта:

1. Не доходя до min(n, m) разряда, оказалось, что i-ые разряды не совпали. Если бы два числа имели одинаковую разрядность, на этом этапе можно было бы заканчивать сравнение, однако нужно убедиться, что число, i-ый разряд которого оказался больше i-го разряда второго числа, не имеет меньшую разрядность, чем второе число. Для этого надо сравнить, сколько разрядов сохранилось у A, а сколько у B. Проверить это можно через поочередное удаление разрядов то у A, то у B. Удалять будем до тех пор, пока у A или у B не закончатся все разряды. Как закончатся, можно будет ответить, какое число больше другого
2. Мы дошли до min(n, m) разряда, и в этом случае числа либо равны, либо одно имеет большую разрядность, чем второе. Проверка того, какой из этих 2 случаев соответствует действительности, будет заключаться в анализе наличия разрядов между двумя символами x, внутри которых были размещены наши числа. Если обнаружатся разряды у A, значит A больше, обнаружатся у B – B больше, не обнаружатся ни у кого – A = B

# **3. Описание состояний**

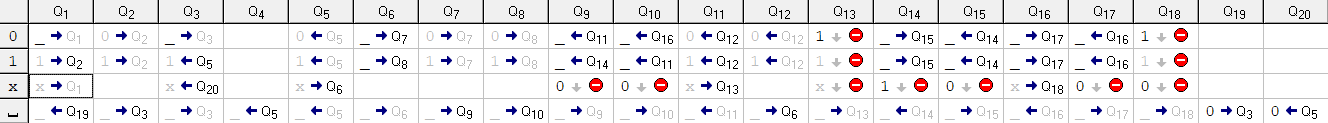
****

Рис. 1 Таблица переходов состояний

Q1 – начальное состояние. Здесь мы удаляем нули до тех пор, пока не встретим единицу или пробел. Встретили единицу – перешли в состояние Q2, в котором просто пройдем по остальным разрядам числа A, после чего перейдём в Q3, встретили пробел – шагнули влево и перешли в Q19, чтобы в Q19 поставить 0 (т.е. это тот случай, когда A = 0), а потом из Q19 перейти в Q3.

Q3 – состояние, при котором мы после избавления числа A от незначащих нулей идём в число B и тоже избавляем его от незначащих нулей. Аналогично, если, находясь в состоянии Q3, мы не встретили в числе B ни одной единицы (о чем будет сигнализировать считывание символа x), мы переходим в Q20, где ставим в B нолик, чтобы B тоже не было пустым, после чего переходим в состояние Q5. Если же мы встретили в B единицу (т.е. B != 0), переходим в состояние Q5.

В Q5 мы просто идем к левому символу x. Как дошли, переходим в Q6. На этом этапе мы отчистили наши числа от незначащих нулей. Далее:

Если, находясь в состоянии Q6, мы встретили 0 или 1, значит, текущий старший разряд числа A равен 0 или 1 соответственно. В зависимости от того, равен этот разряд 0 или 1, мы переходим в Q7 или Q8 соответственно. В состояниях Q7 и Q8 мы просто проходим по остальным разрядам A, разница лишь в том, что, когда мы достигаем пробел, при Q7 мы переходим в Q9, а при Q8 – в Q10.

Таким образом, в Q9 мы попадаем, если текущий старший разряд у A равен 0, а в Q10 – если равен 1.

Если в состоянии Q9 (a1 = 0) мы встретили 0 (то есть b1 = 0) или в состоянии Q10 (a1 = 1) мы встретили 1 (то есть b1 = 1), значит, разряды совпали, и мы переходим и в том, и в другом случае в Q11. В состоянии Q11 мы идем влево до тех пор, пока не встретим младший текущий разряд A. Если мы его не встретили (то есть в состоянии Q11 считали символ x), значит, мы рассмотрели все разряды числа A, и теперь надо проверить, остались ли разряды у числа B. Для этого мы переходим в Q13 и идём там вправо. Если встретили 0 или 1, значит, у B еще есть разряды и B > A, если не встретили, значит, у B нет разрядов, и, так как в Q11, а, следовательно, и в Q13, мы попадаем лишь при условии a1 = b1, получаем, что A = B.

Теперь, если в состоянии Q11 встретили младший текущий разряд A (то есть считали 0 или 1), переходим в состояние Q12, в котором идём влево до тех пор, пока не встретим текущий старший разряд A (обращаю внимание, что, когда мы находились в состоянии Q6, после обнаружения предыдущего старшего разряда A мы его сразу удаляли (см. рис. 1)). Как встретили текущий старший разряд A, снова переходим в состояние Q6.

Пришло время пояснить, что произойдёт, если в состоянии Q9 (a1 = 0) встретить 1 (то есть b1 = 1) или в состоянии Q10 (a1 = 1) мы встретить 0 (то есть b1 = 0).

Для начала рассмотрим случай с Q9:

Мы получили, что ai < bi, значит, для определения max(A, B) достаточно сравнить только разрядности чисел, то есть нам уже не надо будет бегать по каждому из оставшихся разрядов, потому что мы точно знаем, что A != B, а также что A будет больше B с случае, если у двух чисел разрядность одинаковая. Поэтому из Q9 при считывании единицы мы перейдем в Q14, в котором будем идти влево до младшего разряда A, удалять его, переходить в Q15, идти вправо до старшего разряда B, удалять его, затем опять переходить в Q14… То есть мы попадём в цикл Q14-Q15-Q14, в котором будем удалять то разряд у A, то разряд у B. В зависимости от того, в каком состоянии (Q14 или Q15) мы встретим символ x, легко сказать, какое число максимальное.

В случае с Q10 чуть сложнее:

Мы получили, что ai > bi, значит, при определении max(A, B) можно исходить из тех же суждений, что и при Q9 (ai < bi). Собственно, здесь так и сделано – если Вы посмотрите на рис. 1, то увидите, что столбцы Q14 и Q15 аналогичны столбцам Q16 и Q17. Единственное отличие – это переход в Q18 из состояния Q16 при считывании x. Причина у этого следующая:

Предположим, мы сравниваем пару числе x11\_10x. Состояние ленты будет меняться следующим образом:

x11\_10x a1 = ? b1 = ?

x\_1\_10x a1 = 1 b1 = ?

x\_1\_\_0x a1 = 1 b1 = 1 (разрядность чисел уменьшили, a1 и b1 новые)

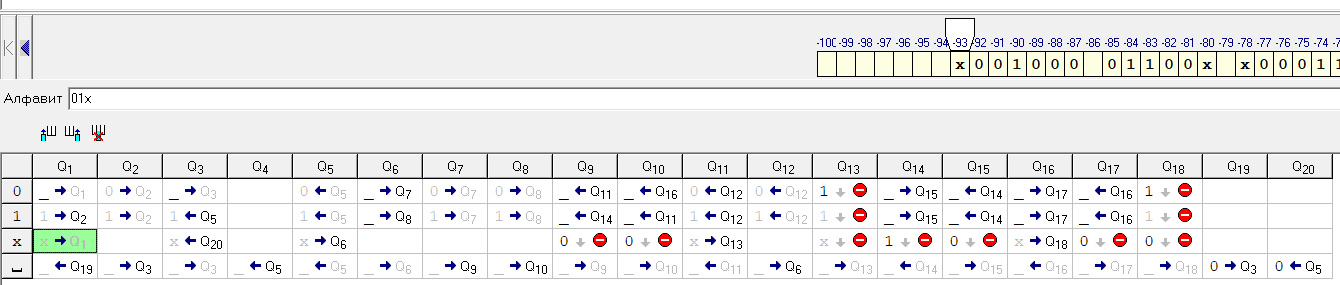
x\_\_\_\_0x a1 = 1 b1 = ?

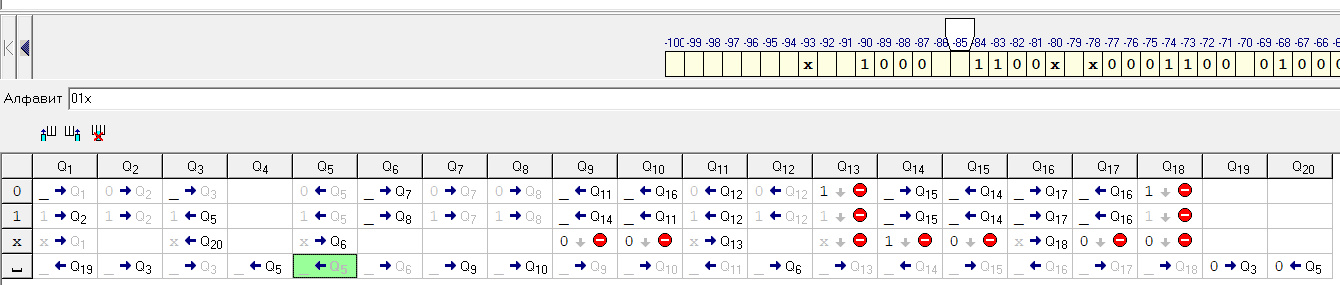
x\_\_\_\_\_x a1 = 1 b1 = 0

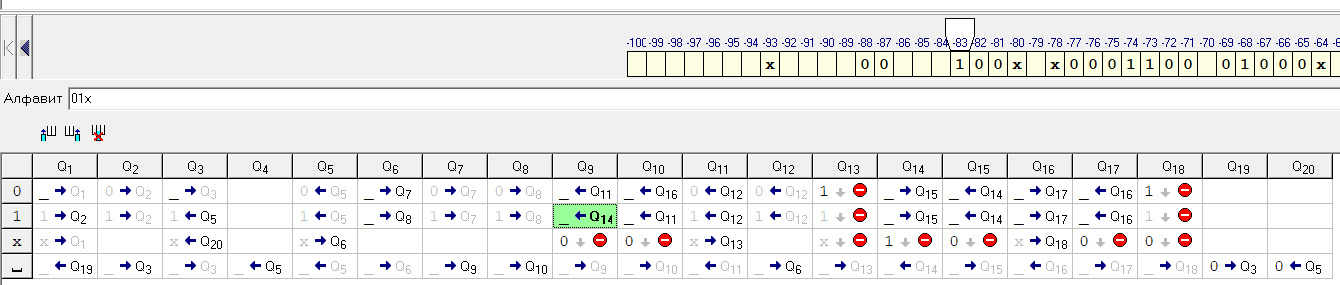
В данном примере мы попадём в Q16, когда головка будет расположена на самой правой ячейке. Если из неё просто пойти влево и при считывании x написать, что B > A, мы получим ошибку. Именно поэтому требуется состояние Q18 – оно позволяет пройти еще и слева-направо, чтобы получить верный ответ.

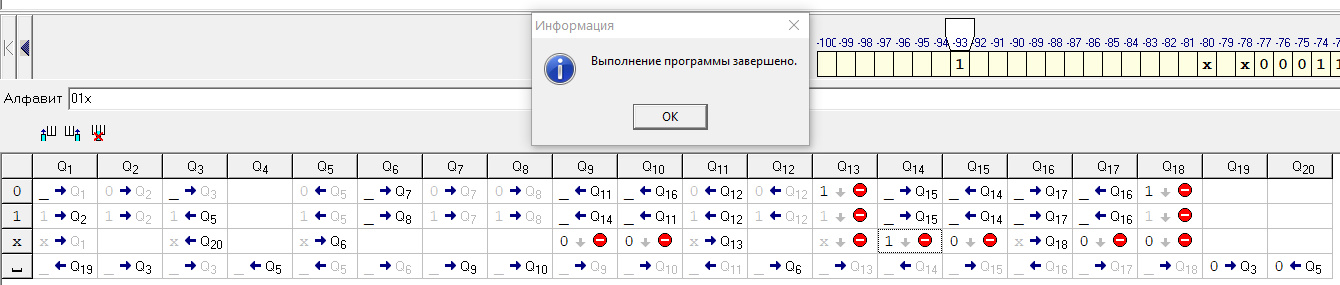
# **4. Работа программы**

Пример работы программы № 1 (x001000\_01100x – результат: 1):

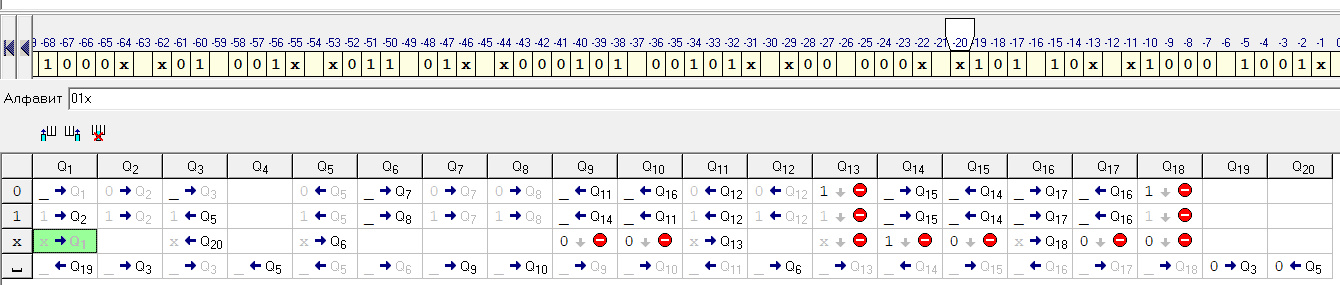


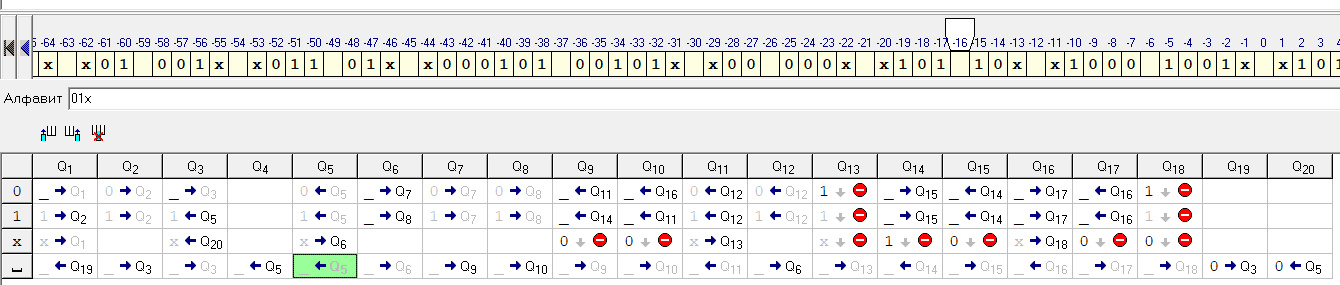


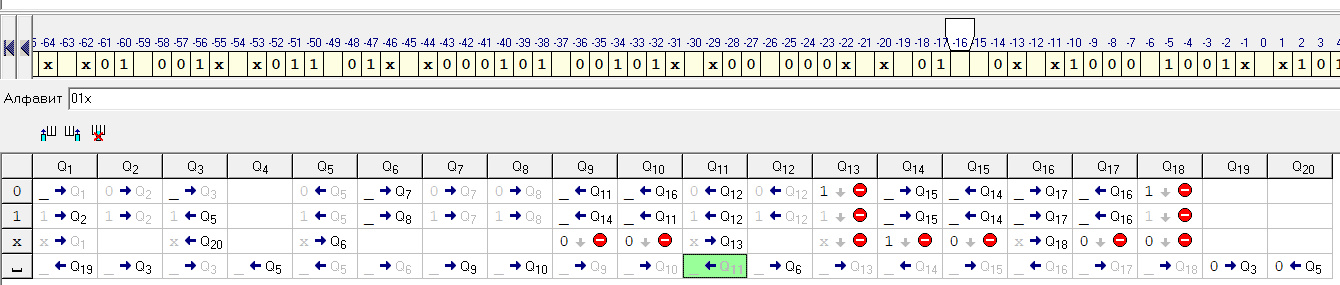


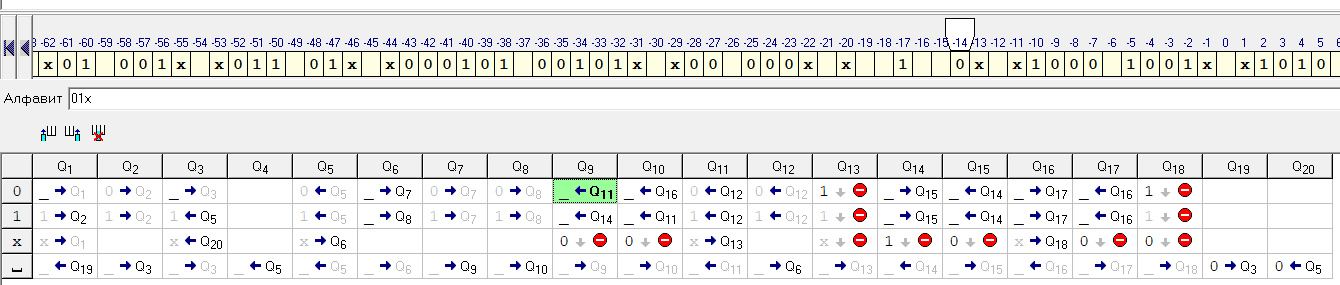


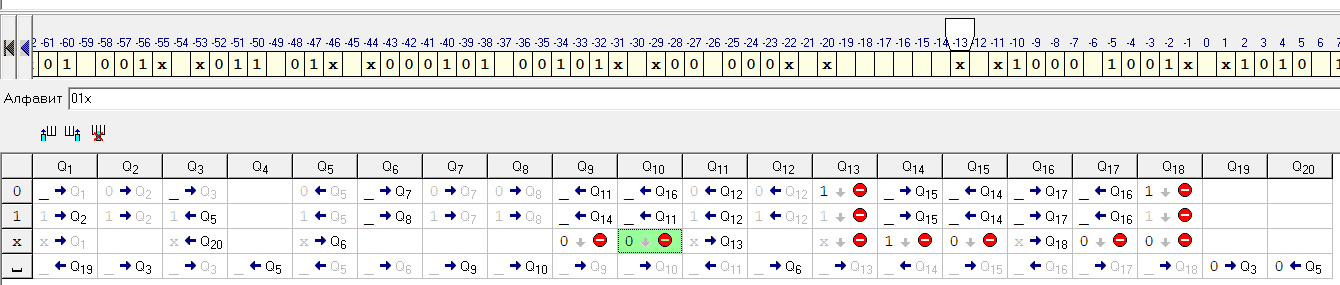
Пример работы программы № 2 (x101\_10x – результат: 0):

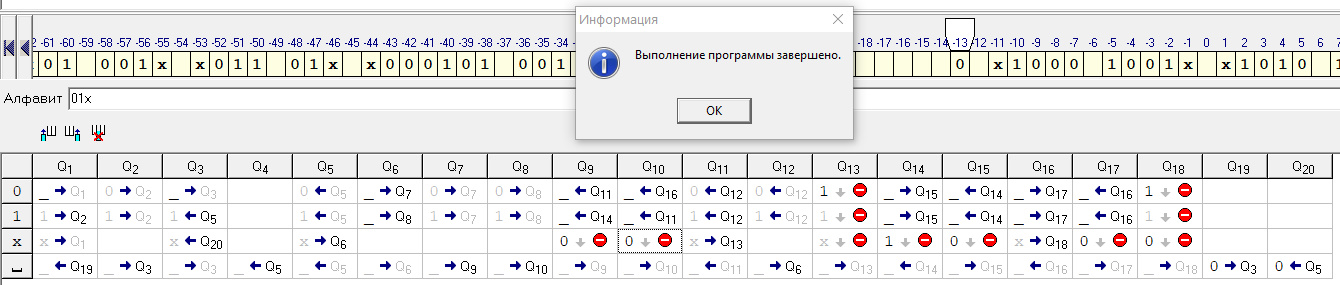












Пример работы программы № 3 (x01\_001x – результат: x):

