Національний технічний університет України «Київський політехнічний інститут»

Науково-навчальний комплекс «Інститут прикладного системного аналізу»

Кафедра математичних методів системного аналізу

***Розрахункова робота №1***

З курсу «Математична логіка і теорія алгоритмів»

На тему «Машина Тьюрінга»

**Виконав:**

Барзій Ілля, КА-41

**Прийняв:**

Спекторський І.Я.

Київ 2015

***Зміст***

Постановка задачі…………………………………………………………………………………………………………………………………3

Обґрунтування алгоритму…………………………………………………………………………………………………………………….3

Схема алгоритму……………………………………………………………………………………………………………………………………3

Список команд………………………………………………………………………………………………………………………………………4

Графічне зображення……………………………………………………………………………………………………………………………7

Табличне зображення……………………………………………………………………………………………………………………………8

Результати роботи програми…………………………………………………………………………………………………………………9

2

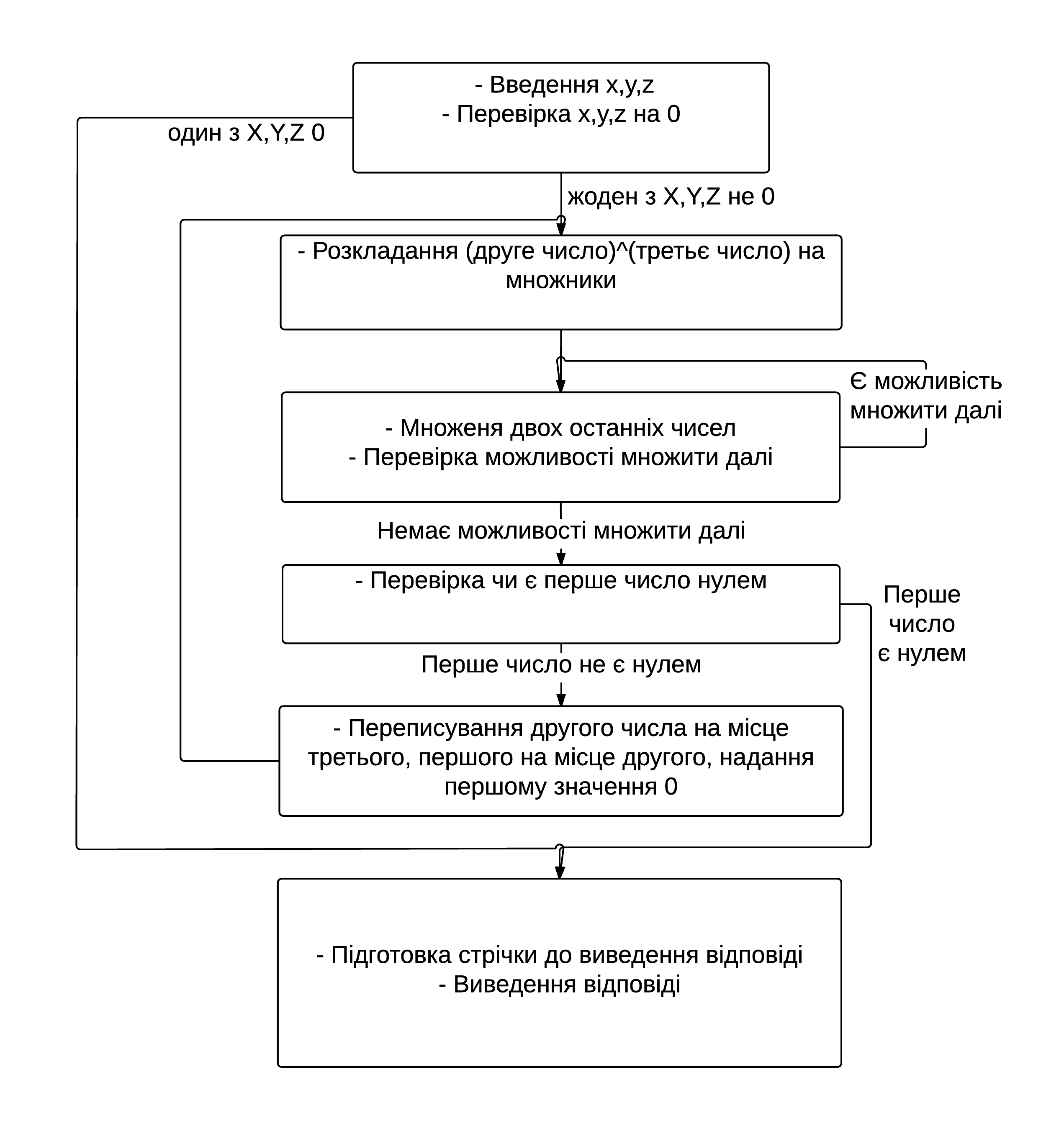
***Постановка задачі***

На вхід дані числа х у z двійковій системі. Необхідно обчислити .

***Обрґунтування алгоритму***

Алгоритм полягає в тому, щоб розкласти число в степені на добуток чисел та порахувати його почерговим множенням. На початку роботи алгоритм перевіряє введену комбінацію чисел на тривіальні випадки (наявні числа 0) і, якщо такі є, одразу ж виводить відповідь. Інакше, розкладає спочатку y^z як добуток та рахує попарно. Потім зводить стрічку до вигляду, коли можна застосувати цю ж саму дію для x^(y^z) та виконує її. Після чого виводить результат.

***Схема алгоритму***



* Введення числа
* іднесення його до квадрату

3

***Список команд***

***Машина перевірки нулів та одиниць***

*start 0 \_ r s0*

*start 1 1 r s1*

*s0 \_ 1 \* !*

*s0 \* \_ r s0*

*s1 ^ ^ r s2*

*s1 \* \* r s1*

*s2 0 0 r s20*

*s2 1 1 r s21*

*s20 \_ \_ l s200*

*s20 \* \_ r s20*

*s200 \_ \_ l s200*

*s200 \* \_ l s201*

*s201 \_ 1 \* !*

*s201 \* \_ l s201*

*s21 ^ ^ r s22*

*s21 \* \* r s21*

*s22 1 1 l s221*

*s22 0 \_ l s220*

*s221 \_ \_ r p1*

*s221 \* \* l s221*

*s220 ^ \_ l s2200*

*s2200 ^ \_ l s2201*

*s2200 \* \_ l s2200*

*s2201 \_ \_ r !*

*s2201 \* \* l s2201*

***Машина розкладання степеня у множники***

*p1 ^ ^ r 01*

*p1 \* \* r p1*

*01 1 1 r 001*

*01 0 \_ l 000*

*000 \_ \_ r 0000*

*000 \* \_ l 000*

*0000 ^ \_ r !*

*0000 \* \* r 0000*

*001 ^ ^ l 02*

*001 \* \* l o1*

*o1 \_ \_ r o2*

*o1 \* \* l o1*

*; if 02 you must rewrite*

*02 1 \_ r 03*

*03 ^ \_ r 0c*

*0c \_ \_ l g1*

*0c 0 8 l 0c0*

*0c 1 7 l 0c1*

*0c0 ^ ^ r 0c00*

*0c0 1 1 r 0c00*

*0c0 0 0 r 0c00*

*0c0 \* \* l 0c0*

*0c1 ^ ^ r 0c11*

*0c1 1 1 r 0c11*

*0c1 0 0 r 0c11*

*0c1 \* \* l 0c1*

*0c00 \_ 0 r 0e*

*0c11 \_ 1 r 0e*

*0e 7 \_ r 0c*

*0e 8 \_ r 0c*

*0e \* \* r 0e*

*g1 ^ ^ l g2*

4

*g1 \* \* l g1*

*g2 \_ ^ l g3*

*g2 \* \* l g2*

*g3 \_ 0 \* end4*

*; Unchanged road*

*o2 \* \* l 1p2*

*1p2 \_ = r 1p3*

*1p3 ^ ^ r 1p4*

*1p3 \* \* r 1p3*

*1p4 ^ ^ r 1p5*

*1p4 \* \* r 1p4*

*1p5 \_ \_ l 1pa*

*1p5 \* \* r 1p5*

*; Here Ive reached the last symbol and begin to rewrite*

*1pa 0 0 l 1pa*

*1pa 1 0 r 1pa1*

*1pa ^ ^ l 1pf*

*1pa1 0 1 r 1pa1*

*1pa1 \_ \_ l 1pa2*

*1pa2 ^ ^ l 1pad*

*1pa2 \* \* l 1pa2*

*;8 is an alternative 0 7 is an alternative 1*

*1pad 0 8 r 1pad0*

*1pad 1 7 r 1pad1*

*1pad0 \_ 0 r 1padr*

*1pad0 \* \* l 1pad0*

*1pad1 \_ 1 r 1padr*

*1pad1 \* \* l 1pad1*

*1padr 8 0 l 1pad\*?*

*1padr 7 1 l 1pad\*?*

*1padr \* \* r 1padr*

*1pad\*? ^ ^ l 1pad\**

*1pad\*? \* \* \* 1pad*

*1pad\* \_ # r 1padb*

*1pf2 = = l 1pf0*

*1pad\* \* \* l 1pad\**

*1padb \_ \_ l 1pa*

*1padb \* \* r 1padb*

*1pf \_ \_ r 1pf1*

*1pf \* \* l 1pf*

*1pf1 # = r 1pf2*

*1pf2 # # r 1pf3*

*1pf2 \* \* r 1pf2*

*1pf3 # # l end1*

*1pf3 = = l end1*

*1pf3 \* \* r 1pf3*

*1pf0 = \_ r end32*

*1pf0 \* \* l 1pf0*

***Машина множення***

*end1 1 7 l q1*

*end1 0 8 l 15*

*end1 # # l 17*

*;Describing line of 0*

*15 \_ \_ \* 16*

*15 7 7 l 16*

*15 8 8 l 16*

*15 \* \* l 15*

*16 1 7 r 13*

*16 0 8 r 13*

*16 \_ 8 r 13*

*16 \* \* l 16*

*;Describing line of 1*

*q1 # # l 2*

*q1 \* \* l q1*

*2 = = l 10*

*2 1 7 l 3*

*2 0 8 l 6*

*; Describing == line in 2*

*10 9 0 l 10*

*10 6 1 l 10*

*10 7 7 l 10*

*10 8 8 l 10*

*10 \* \* r 11*

*11 1 1 r 11*

*11 0 0 r 11*

*11 \* \* l 12*

*12 0 8 r 13*

*12 1 7 r 13*

*13 = = r 14*

*13 \* \* r 13*

*14 1 1 r 14*

55

*14 0 0 r 14*

*14 # # r 14*

*14 \* \* \* 1*

*1 7 1 l end1*

*1 8 0 l end1*

*;Describing 1 line in 2*

*3 = = l 4*

*3 \* \* l 3*

*4 1 9 l 5*

*4 \_ 6 r 8*

*4 0 6 r 8*

*4 \* \* l 4*

*5 1 0 l 5*

*5 0 1 r 8*

*5 \_ 1 r 8*

*8 = = r 9*

*8 \* \* r 8*

*9 7 1 l 2*

*9 8 0 l 2*

*9 \* \* r 9*

*; Describing 0 line in 2*

*6 = = l 7*

*6 \* \* l 6*

*7 \_ 9 r 8*

*7 0 9 r 8*

*7 1 6 r 8*

*7 \* \* l 7*

*;Describing exit from 17*

*17 \_ \_ r end2*

*17 7 1 l 17*

*17 8 0 l 17*

*17 \* \* l 17*

***Машина перенесення множника та вибору подальших дій***

*end2 \* \* l 3q1*

*3q1 \_ = r 3q2*

*3q2 # # r 3q3*

*3q2 \* \* r 3q2*

*3q3 = = l 3q4*

*3q3 # # l 3q4*

*3q3 \* \* r 3q3*

*3q4 = \_ l 3q5*

*3q4 \* \_ l 3q4*

*3q5 1 \_ r 3q51*

*3q5 0 \_ r 3q50*

*3q51 \_ \_ r 3q51*

*3q51 \* \* l 3q53*

*3q53 \_ 1 l 3qe*

*3q50 \_ \_ r 3q50*

*3q50 \* \* l 3q52*

*3q52 \_ 0 l 3qe*

*3qe 1 1 \* 3q5*

*3qe 0 0 \* 3q5*

*3qe \_ \_ l 3qe*

*3qe = \_ r 3qe2*

*3qe2 1 1 l 3qe3*

*3qe2 0 0 l 3qe3*

*3qe2 \* \* r 3qe2*

*3qe3 \_ = r 3qe4*

*3qe4 # # r 3qen1*

*3qe4 = = l 3qen2*

*3qe4 \* \* r 3qe4*

*3qen1 # # l end31*

*3qen1 = = l end31*

*3qen1 \* \* r 3qen1*

*3qen2 = \_ r end32*

*3qen2 \* \* l 3qen2*

***Машина вибору подальших дій***

*end32 = = r tq=*

*end32 \* \* r end32*

*tq= 0 0 l tq0*

*tq= 1 1 l tq1*

*tq1 \_ \_ r endp1*

*tq1 \* \* l tq1*

*tq0 \_ \_ r endp2*

*tq0 \* \* l tq0*

***Машина підготовки стрічки до повторного піднесення до степеня***

*endp1 \* \* l 4q1*

*4q1 \_ = r 4q2*

*4q2 ^ ^ r 4q3*

*4q2 \* \* r 4q2*

*4q3 \_ \_ l 4q4*

*4q3 \* \_ r 4q3*

6

*4q4 = + l 4q5*

*4q4 \* \* l 4q4*

*4q5 = = r 4qp*

*4q5 \* \* l 4q5*

*4qp + = l 4qd*

*4qp 8 8 r 4qp*

*4qp 7 7 r 4qp*

*4qp 0 8 r 4qp0*

*4qp 1 7 r 4qp1*

*4qp1 \_ 1 l 4q5*

*4qp1 \* \* r 4qp1*

*4qp0 \_ 0 l 4q5*

*4qp0 \* \* r 4qp0*

*4qd \_ \_ r 4qd1*

*4qd \* \_ l 4qd*

*4qd1 = ^ l 4qd2*

*4qd1 \_ \_ r 4qd1*

*4qd2 \_ 0 \* end4*

***Машина підготовки стрічки до виводу відповіді***

*endp2 \* \* l eo1*

*eo1 \_ = r eo2*

*eo2 = \_ r eq1*

*eo2 \* \* r eo2*

*eq1 \_ \_ l eq2*

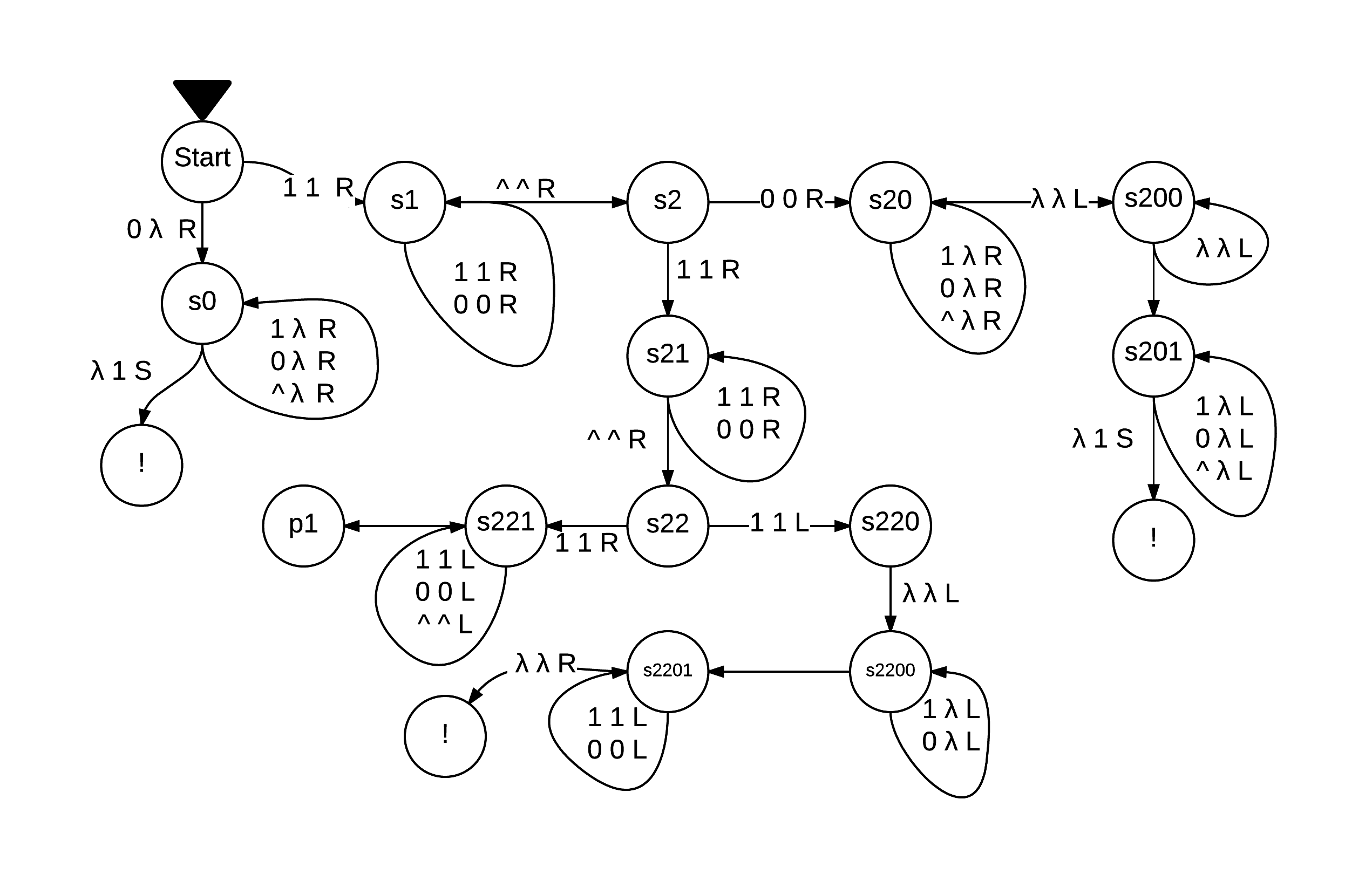
*eq1 \* \_ r eq1*

*eq2 = \_ r !*

*eq2 \* \* l eq2*

***Графічне зображення***

Зображення машини Тьюрінга у вигляді орієнтованого міченого графу зобразимо на прикладі перевірки x,y,z на наявність 0.



7

***Табличне зображення***

Зображення машини Тьюрінга у вигляді таблиці покажемо на прикладі перевірки x,y,z на наявність 0. 

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 0 | ^ | λ |
| start | λ s0 R | 1 s1 R |  |  |
| s0 | λ s0 R | λ s0 R | λ s0 R | 1 ! S |
| s1 | 1 s1 R | 0 s1 R | ^ s2 R |  |
| s2 | 1 s21 R | 0 s20 R |  |  |
| s20 | λ s20 R | λ s20 R | λ s20 R | λ s200 L |
| s200 | 1 s201 λ | 0 s201 λ | ^ s201 λ | λ s200 L |
| s201 | λ s201 L | λ s201 L | λ s201 L | 1 ! S |
| s21 | 1 s21 R | 0 s21 R | ^ s22 R |  |
| s22 | 1 s221 L | 0 s220 L |  |  |
| s221 | 1 s221 L | 0 s221 L | ^ s221 L | λ p1 R |
| s220 |  |  |  | λ s2200 L |
| s2200 | λ s2200 L | λ s2200 L | λ s2201 L |  |
| s2201 | 1 s2201 L | 0 s2201 L |  | λ ! R |
| p1 |  |  |  |  |
| ! |  |  |  |  |

8

***Результати роботи програми***

***Вхідні дані: 111^11^0***

***Вихідні дані: 111***

***Кількість команд: 17***

***Вхідні дані: 101^0^11***

***Вихідні дані: 1***

***Кількість команд: 11***

***Вхідні дані: 10^10^10***

***Вихідні дані: 10000***

***Кількість команд: 1436***

***Вхідні дані: 101^11^11***

***Вихідні дані: 10011101100101110001111001001111111010000100000000011110011101***

***Кількість команд: 260529***

9