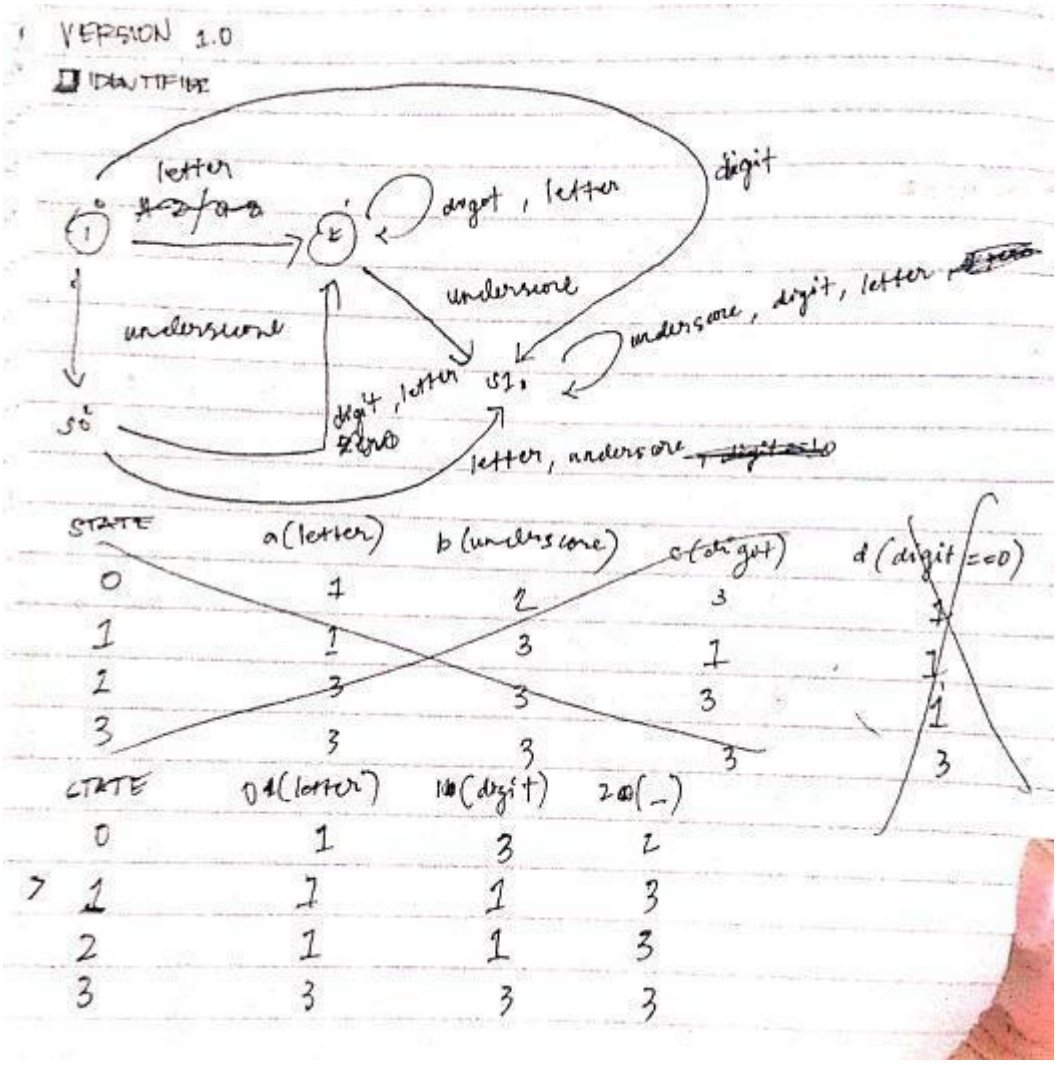
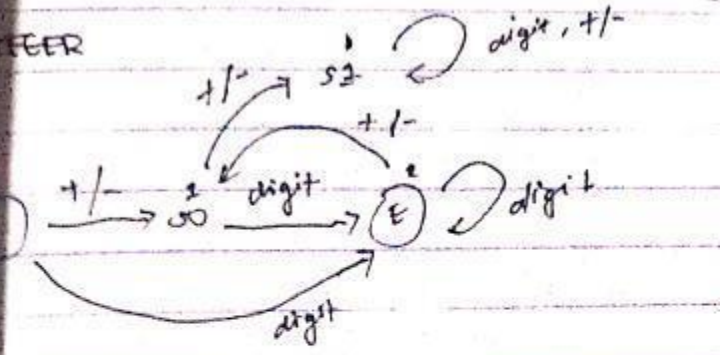
Version 1.0

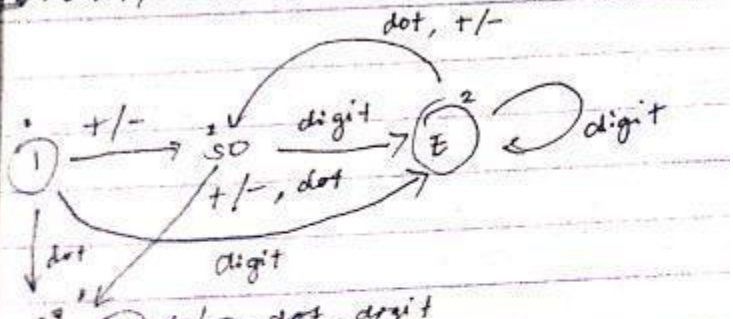

VERSION 1.0
IDENTIFIER
letter
digit
digit , letter
underscore
underscore, digit, letter
underscore
digit, letter
letter, underscore
STATE
a(letter)
b(underscore)
c(digit)
d(digit!=0)
CTATE
0a(letter)
1b(digit)
2c(_)

1) INTEGER



STATE	a(+/-)	b(digit)
0	1	2
1	3	2
2	1	2
3	3	3

2) FLOAT/REAL NUMBER



STATE	a(+/-)	b(digit)	c(dot)
0	1	2	3
1	3	2	3
2	1	2	1
3	3	3	3

```java
static boolean isIdentifier(String str)
{
    int state = 0, flag = 0, input = 0;
    int[][] table = {{1, 3, 2}, {1, 1, 3}, {1, 1, 3}};

    for(char symbol: str.toCharArray())
    {
        if(Character.isAlphabetic(symbol))
            flag = 0;
        else if(Character.isDigit(symbol))
            flag = 1;
        else if(symbol == '_')
            flag = 2;

        switch(flag)
        {
            case 0: input = 0;
                    break;
            case 1: input = 1;
                    break;
            case 2: input = 2;
                    break;
        }

        state = table[state][input];
    }

    if(state == 1)
        return true;
    else
        return false;
}
```

```java
static boolean isInteger(String str)
{
    int state = 0, x = 0, input = 0;
    int[][] table = {{1, 2}, {3, 2}, {1, 2}, {3, 3}};

    for(char symbol: str.toCharArray())
    {
        if(symbol == '+' || symbol == '-')
            x = 0;
        else if(Character.isDigit(symbol))
            x = 1;

        switch(x)
        {
            case 0: input = 0;
                    break;
            case 1: input = 1;
                    break;
        }

        state = table[state][input];
    }

    if(state == 3)  return true;
    else    return false;
}
```

```java
static boolean isFloat(String str)
{

    int state = 0, x = 0, input = 0;
    int[][] table = {{1, 2, 3},{3, 2, 3},{1, 2, 1},{3, 3, 3}};

    for(char symbol: str.toCharArray())
    {
        if(symbol == '+' || symbol == '-')
            x = 0;
        else if(Character.isDigit(symbol))
            x = 1;
        else if(symbol == '.')
            x = 2;

        switch(x)
        {
            case 0: input = 0;
                    break;
            case 1: input = 1;
                    break;
            case 2: input = 2;
                    break;
        }

        state = table[state][input];
    }

    if(state == 2)  return true;
    else    return false;
}
```