Data Structure Homework 3

Student ID : 41247001S

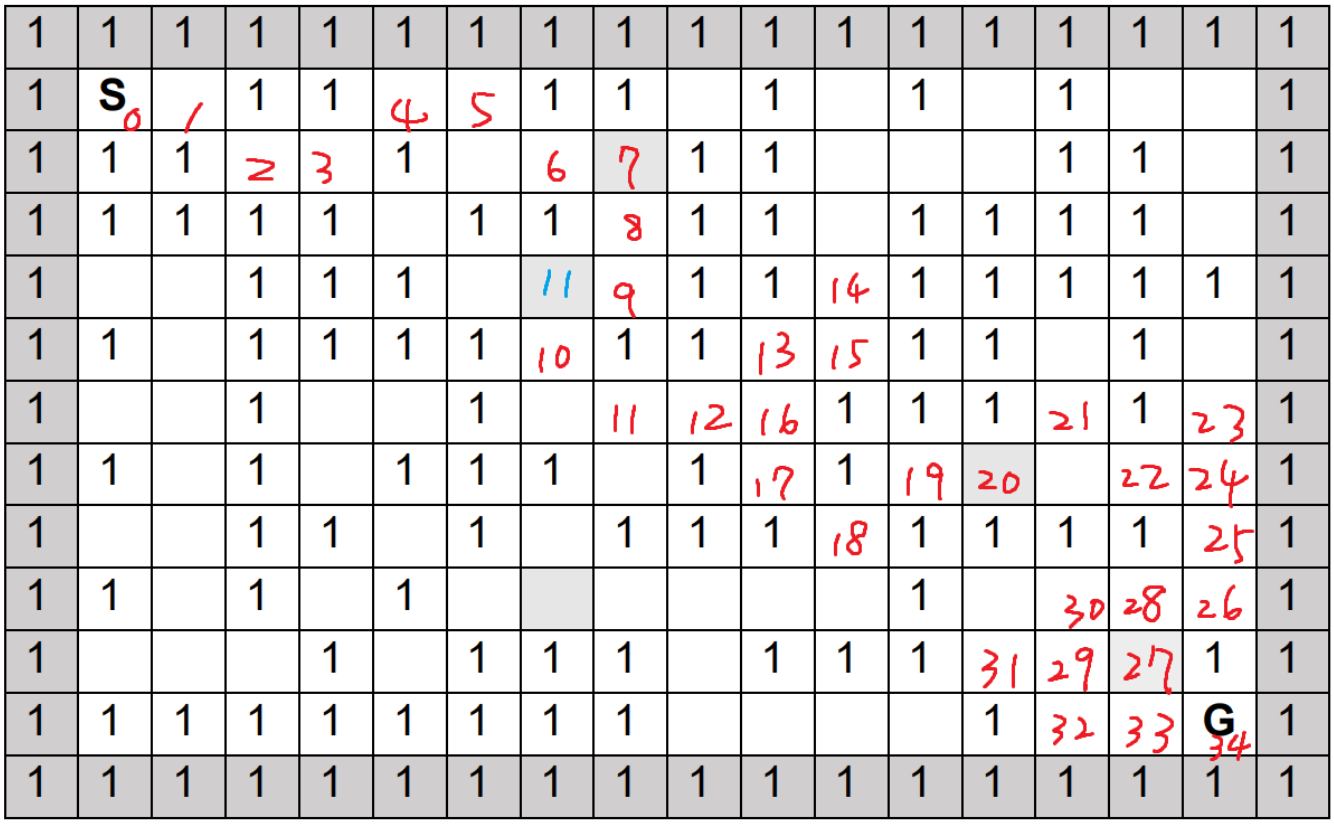
Question 1 (a)

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	S 1	2	1	1	5	6	1	1	9	1	23	1	25	1			1
1	1	1	3	4	1	16	7	8	1	1	22	24	26	1	1		1
1	1	1	1	1	15	1	1	10	1	1	21	1	1	1	1		1
1			1	1	1	14	13	11	1	1	20	1	1	1	1	1	1
1	1		1	1	1	1	12	1	1	19	27	1	1	34	1	37	1
1			1			1		17	18	28	1	1	1	33	1	36	1
1	1		1		1	1	1		1	29	1	31	32		35	38	1
1			1	1		1		1	1	1	30	1	1	1	1	39	1
1	1		1		1							1	46	44	42	40	1
1				1		1	1	1		1	1	1	45	43	41	1	1
1	1	1	1	1	1	1	1	1					1	47	48	G 49	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Question 1(b)

Cell	i (of stack [i])	< x_i, y_i, *dir* >
(2, 8)	7	2, 8, 4
(4, 7)	11	4, 7, 6
(7, 13)	20	7, 13, 1
(9, 7)	X	X
(10, 15)	27	10, 15, 0

Answer derived from the below path.



Question 2(a)

(a) 8 * (4 + 2) / (9 - (3 + 4)) * (3 / (5 - 2))

= 8 x 4 2 + / (9 - 3 4 +) x (3 / 5 2 -)

= 8 4 2 + x / 9 3 4 + - x 3 5 2 - /

= 8 4 2 + x 9 3 4 + - / 3 5 2 - / x

Stack:

Question 2(b)

(b) ((a * (b – k)) + m) / (((n - p) * n) * (d + e))

= ((a x bk-)+m)/((np-xn)xde+)

= (abk-x+m)/(np-nxxde+)

= abk-x m+/ np-n xde+x

= abk-xm+np-nxde+x/

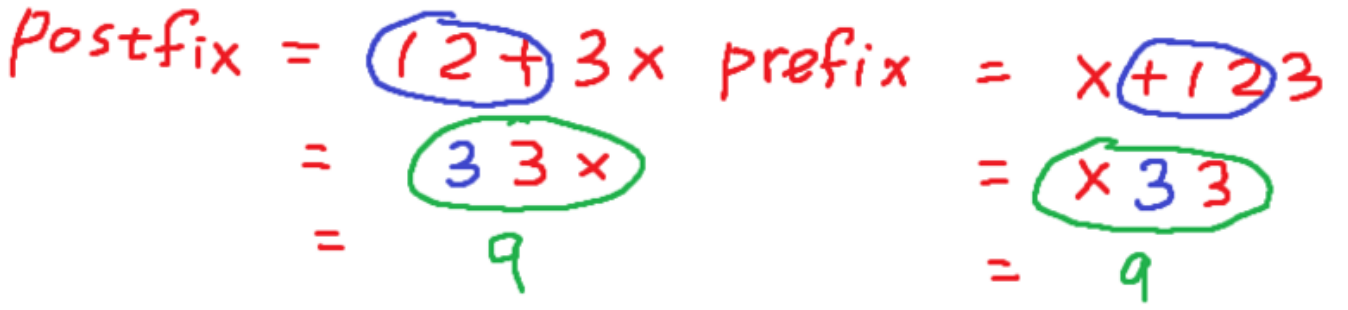
postfix	stack
a	a
ab	a , b
abk	a , b , k
abk-	a ,(b-k)
abk-*	a*(b-k)
abk-*m	a*(b-k) , m
abk-*m+	a*(b-k) + m
abk-*m+n	a*(b-k) + m , n
abk-*m+np	a*(b-k) + m , n , p
abk-*m+np-	a*(b-k) + m , n-p
abk-*m+np-n	a*(b-k) + m , n-p , n
abk-*m+np-n*	a*(b-k) + m , (n-p)*n
abk-*m+np-n*d	a*(b-k) + m , (n-p)*n , d
abk-*m+np-n*de	a*(b-k) + m , (n-p)*n , d , e
abk-*m+np-n*de+	a*(b-k) + m , (n-p)*n , (d+e)
abk-*m+np-n*de+*	a*(b-k) + m , ((n-p)*n)*(d+e)
abk-*m+np-n*de+*/	(a*(b-k) + m) / (((n-p)*n)*(d+e))

Question 3

postfix	stack
1	1
12	1,2
12*	2
12*3	2,3
12*34	2,3,4
12*34+	2,7
12*34+5	2,7,5
12*34+56	2,7,5,6
12*34+567	2,7,5,6,7
12*34+567+	2,7,5,13
12*34+567+8	2,7,5,13,8
12*34+567+89	2,7,5,13,8,9
12*34+567+89-	2,7,5,13,-1
12*34+567+89-+	2,7,5,12
12*34+567+89-+*	2,7,60
12*34+567+89-+*+	2,67
12*34+567+89-+*+-	-65

Question 4

因為題目從 postfix 改成 prefix，↵
所以判斷做運算的規則要改變↵
postfix 是判斷到有運算符號（加減乘除）的時候↵
去把 stack 最上面的兩個元素做運算後放到 stack↵
而 prefix 的話會變成我們當目前的 token 是數字↵
且 stack.top() 也是數字時，去做 stack 由上數來第二個的運算符號↵
↵




```
# pseudo postfix eval code
# this is based on code in homework 3
postfix:
    while(token != endOfString):
```

```
        if(token.isdigit()) stack.push(int(token))
        else:
            number2 = stack.pop()
            number1 = stack.pop()
            stack.push(calculate(number1 , token , number2))
        token = get_token_fromString()
    return stack.top()

# modify to prefix version
prefix:
    while(token != endOfString):
        if(token.isdigit()):
            if(stack.top().isdigit()):
                number2 = int(token)
                number1 = stack.pop()
                calc = stack.pop() # +-*/
                stack.push(calculate(number1 , calc , number2))
            else:
                stack.push(int(token))
        else:
            stack.push(token)
        token = get_token_fromString()
    return stack.top()
```