Homework 01

PB20000296 郑滕飞

2.1.a

字母：abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
数字：0123456789
符号：! " # % & ' () * + , −. / :; < = > ? [\ ] ^ _ { | } ~
空白符：空格、水平制表符\t、垂直制表符\v、换行\n、换页\f
不可打印字符：字符串终止\0、警报符\a、退格\b、回车\r

2.3.b

由于 0 与 1 都为 0?1*可表示的句子，原式即可看作(0|1)*，可以表示任何由 01 组成的字符串(含空串)。

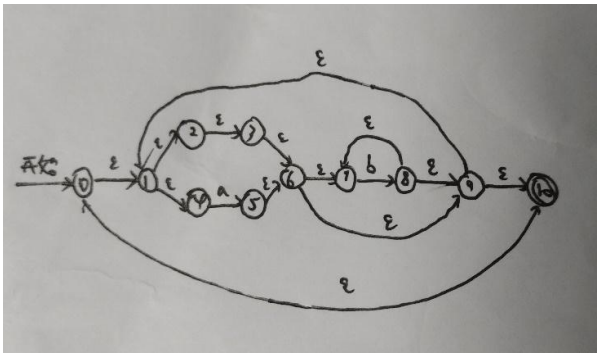
2.4.b

A*a*B*b*C*c*D*d*E*e*F*f*G*g*H*h*I*i*J*j*K*k*L*l*M*m*N*n*O*o*P*p*Q*q*
R*r*S*s*T*t*U*u*V*v*W*w*X*x*Y*y*Z*z*

2.4.i

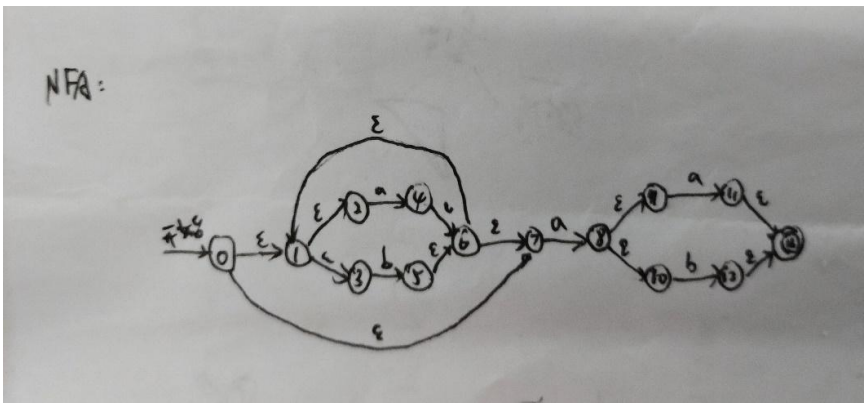
b*(ab⁺)*a?

2.7.c

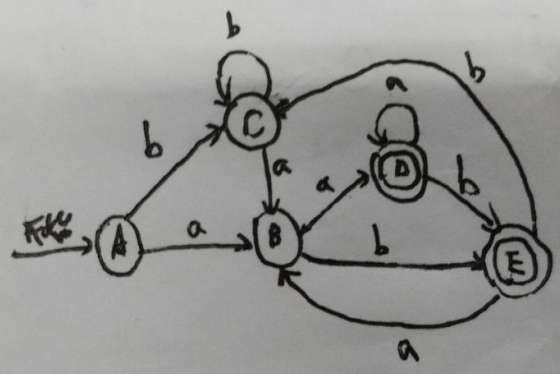


0-1-4-5-6-7-8-9-1-4-5-6-7-8-7-8-9-1-4-5-6-7-8-9-10

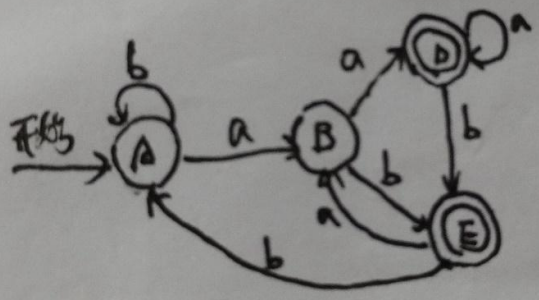
2.12.a



DFA:



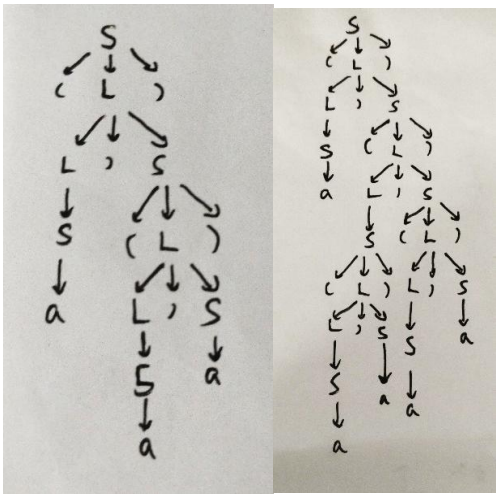
化简DFA:



Homework 02

PB20000296 郑滕飞

3.1.a



3.1.c

S => (L) => (L,S) => (L,(L)) => (L,(L,S)) => (L,(L,a)) => (L,(S,a))
    => (L,(S,a)) => (L,(a,a)) => (S,(a,a)) => (a,(a,a))

S => (L) => (L,S) => (L,(L)) => (L,(L,S)) => (L,(L,a)) => (L,(S,a))
    => (L,(S,a)) => (L,(a,a)) => (S,(a,a)) => ((L),(a,a))
    => ((L,S),(a,a)) => ((L,a),(a,a)) => ((S,a),(a,a)) => ((a,a),(a,a))

3.2.a

S => aSbS => abSaSbS => abaSbS => ababS => abab
S => aSbS => abS => abaSbS => ababS => abab

3.3

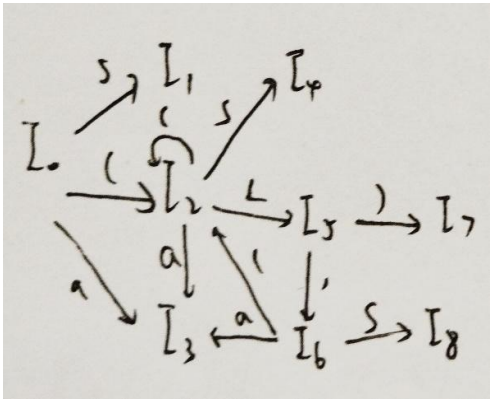
```
S -> S and T | T
T -> T or U | U
U -> not U | true | false | (S)
```

Homework 03

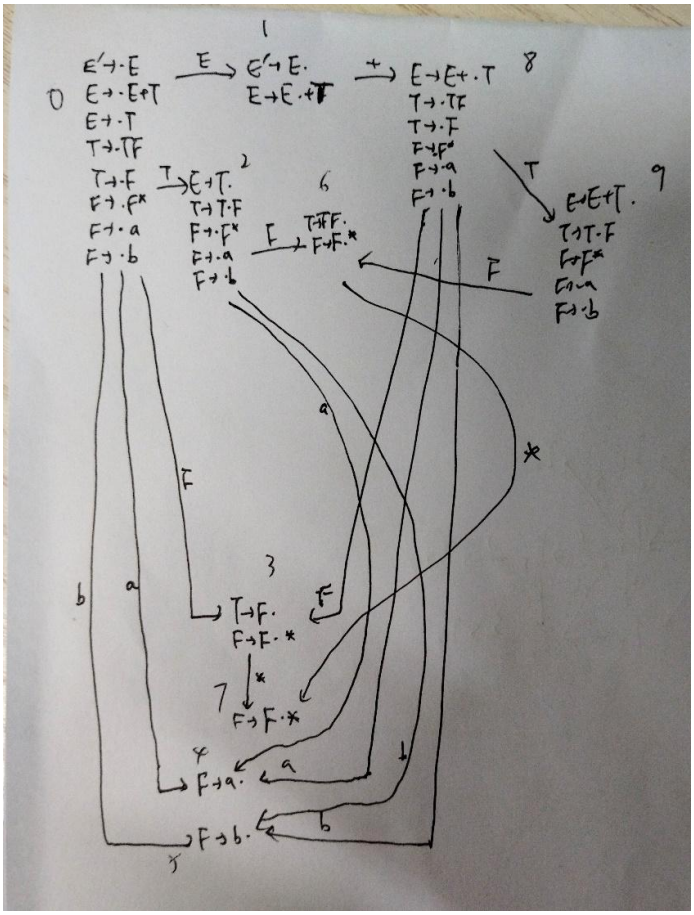
PB20000296 郑滕飞

3.17

(每个状态都可作为终止状态)



3.19.a



1 E->E+T **2** E->T **3** T->TF **4** T->F

5 F->F* **6** F->a **7** F->b

状态	a	b	*	+	$	E	T	F
0	s4	s5				1	2	3
1				s8	acc			
2	s4	s5		r2	r2			6
3	r4	r4	s7	r4	r4			
4	r6	r6	r6	r6	r6			
5	r7	r7	r7	r7	r7			
6	r3	r3	s7	r3	r3			
7	r5	r5	r5	r5	r5			
8	s4	s5					9	3
9	s4	s5		r1	r1			6

3.21.a

是 LL(1)：

FIRST(AaBb)={a} FIRST(BbBa)={b}

FIRST(A)={ε} FOLLOW(B)={a,b}

FIRST(B)={ε} FOLLOW(A)={a,b}

由此满足 LL(1)。

不是 SLR(1)：空串可能按 A 或 B 进行规约，无法确定。

Homework 04

PB20000296 郑滕飞

3.4.b

```
U -> U'|'T | T
T -> TS | S
S -> S* | R
R -> a | b | (U)
```

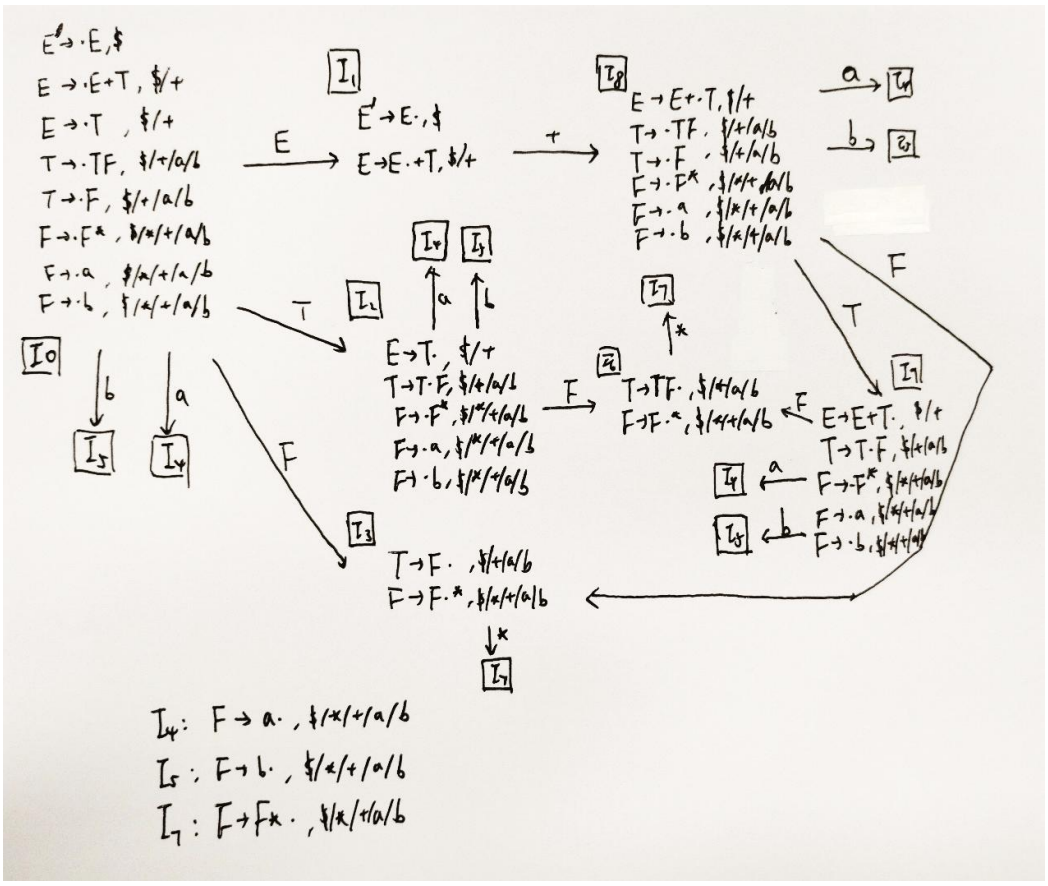
3.10

非终结符	输入符号				
	int	**real**	**id**	**,**	**$**
D	D -> TL	D -> TL			
T	T -> **int**	T -> **real**			
L			L -> **id** R		
R				R -> **,** **id** R	R -> ε

3.12

FIRST(AB) ∩ FIRST(PQx) = {x}，从而不是。

3.19.b



1 E->E+T **2** E->T **3** T->TF **4** T->F

5 F->F* **6** F->a **7** F->b

状态	a	b	*	+	$	E	T	F
0	s4	s5				1	2	3
1				s8	acc			
2	s4	s5		r2	r2			6
3	r4	r4	s7	r4	r4			
4	r6	r6	r6	r6	r6			
5	r7	r7	r7	r7	r7			
6	r3	r3	s7	r3	r3			
7	r5	r5	r5	r5	r5			
8	s4	s5					9	3
9	s4	s5		r1	r1			6

3.27.b

不是 LR(1)。假设第一个字符为 d，则该文法在最开始就需要决定移进 d 还是空规约出 W(否则移进 d 后如果遇到 p，由于只有 W -> Wd，已经来不及在栈中规约出 W)，但这是无法确定的。

Homework 05

PB20000296 郑滕飞

4.3.a

产生式	语义规则
S’ -> S	print(S.val)
S -> (L)	S.num = L.num + 1
S -> a	S.num = 0
L -> L1, S	L.num = L1.num + S.num
L -> S	L.num = S.num

4.3.b

产生式	语义规则
S’ -> S	print(S.val)
S -> (L)	S.maxd = L.maxd + 1
S -> a	S.maxd = 0
L -> L1, S	L.maxd = max(L1.maxd, S.maxd)
L -> S	L.maxd = S.maxd

4.9.b

*修改文法

产生式	语义规则
S -> L.R	S.sum = L1.sum + L2.sum L.pos = 0 R.pos = -1
S -> L	S.sum = L.sum L.pos = 0
L -> L1B	L.sum = L1.sum + B.val L1.pos = L.pos + 1 B.pos = L.pos
L -> B	L.sum = B.val B.pos = L.pos
R -> BR1	R.sum = R1.sum + B.val R1.pos = R.pos − 1 B.pos = R.pos
R -> B	R.sum = B.val B.pos = R.pos
B -> 0	B.val = 0
B -> 1	B.val = pow(2, B.pos)

4.12.a

产生式	语义规则
B -> S	S.depth = 0

S -> (L)	L.depth = S.depth + 1
S -> a	print(S.depth)
L -> L1, S	S.depth = L1.depth = L.depth
L -> S	S.depth = L.depth

```
B -> {S.depth = 0} S
S -> ( {L.depth = S.depth + 1} L)
S -> a {print(S.depth)}
L -> {L1.depth = L.depth} L1, {S.depth = L.depth} S
L -> {S.depth = L.depth} S
```

4.12.b

产生式	语义规则
B -> S	S.pos = 1
S -> (L)	S.len = L.len + 2 L.pos = S.pos + 1
S -> a	S.len = 1 print(S.pos)
L -> L1, S	L.len = L1.len + S.len + 1 L1.pos = L.pos S.pos = L1.len + L1.pos + 1
L -> S	L.len = S.len S.pos = L.pos

```
B -> {S.pos = 1} S
S -> ( {L.pos = S.pos + 1} L) {S.len = L.len + 2}
S -> a {S.len = 1; print(S.pos)}
L -> {L1.pos = L.pos} L1,
    {S.pos = L1.len + L1.pos + 1} S { L.len = L1.len + S.len + 1}
L -> {S.pos = L.pos} S {L.len = S.len}
```

Homework 06

PB20000296 郑滕飞

7.1.d

```
t1 = a + b
t2 = -t1
t3 = c + d
t4 = t2 * t3
t5 = a + b
t6 = t5 + c
t7 = t4 + t6
```

(t7 为结果)

7.2.c

```
t1 = a
t2 = 0
goto JUDGE

JUDGE:
if t2 <= 10 goto COUNT
goto OUT

COUNT:
t1[t2] = 0
goto JUDGE

OUT:
return
```

7.5

```
P -> {D.offset = 0} D; S
D -> {D1.offset = D.offset} D1;
{D2.offset = D1.offset + D1.width} D2
{D.width = D1.width + D2.width}
D -> id:T {enter(id.lexeme), T.type, D.offset); D.width = T.width}
T -> integer {T.type = integer; T.width = 4}
T -> real {T.type = real; T.width = 8}
T -> array[num] of T1
{T.type = array(num.val, T1.type); T.width = num.val * T1.width}
T -> ^T1 {T.type = pointer(T1.type); T.width = 4}
```

Homework 07

PB20000296 郑滕飞

9.3.a

*此处数字代表对应的语句

	gen	kill	IN	OUT
B1	1,2	8,10,11		1,2
B2	3,4	5,6	1,2,3,4,5,8,9	1,2,3,4,8,9
B3	5	4,6	1,2,3,4,6,7,8,9	1,2,3,5,7,8,9
B4	6,7	4,5,9	1,2,3,5,7,8,9	1,2,3,6,7,8
B5	8,9	2,7,11	1,2,3,4,5,7,8,9	1,3,4,5,8,9
B6	10,11	1,2,8	1,3,4,5,8,9	3,4,5,9,10,11

9.3.b

*U = {1, 2, a+b, c-a, b+d, e+1, b*d, a-d}，按序编号为 A 到 H

	e_gen	e_kill	IN	OUT
B1	A, B	C, D, E, G, H		A, B
B2	C, D	E, G, H	A, B	A, B, C, D
B3		E, G, H	A, B, C, D	A, B, C, D
B4	C	E, F, G, H	A, B, C, D	A, B, C, D
B5	D	C, E, F, G	A, B, C, D	A, B, D
B6	H	C, D, E, G	A, B, D	A, B, H

9.3.c

	use	def	IN	OUT
B1		a, b	e	a, b, e
B2	a, b	c, d	a, b, e	a, b, c, d, e
B3	b, d		a, b, c, d, e	a, b, c, d, e
B4	a, b, e	d	a, b, c, e	a, b, c, d, e
B5	a, b, c	e	a, b, c, d	a, b, d, e
B6	b, d	a	b, d	

9.22

```
int i, j, t1 = 0, t2;
int r[20][10];
for (i = 0; i < 20; i++) {
    t2 = 0;
    for (j = 0; j < 10; j++) {
        r[i][j] = t2;
        t2 += t1;
    }
    t1 += 10;
}
```