For the rest of the subjects

I. (4 x 0.5p)

1) Consider $L_1 = \{\varepsilon, 0, 1\}$ and $L_2 = \{0, 00\}$. Which elements belong to $L_1 U L_2$?

a) ε b) 10 c) 00 d) 01

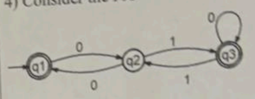
2) Which of the following matches regexp $a(ab)^*a$

a) aaba b) aabbaa c) aba d) aabababa

3) Consider the grammar with productions $S \to RT$; $R \to Ra$; $T \to b$. Which symbols are unproductive?

a) R b) R,S c) R,S,T d) S

4) Consider the FA



Which sequences belong to L(M)?

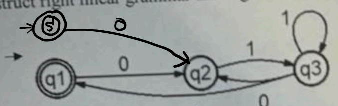
a) 010 b) ε c) 011 d) 000

...corresponding to FA

$a(ab)^*a$

$S \to RT$
$R \to Ra$
$T \to b$

II. (2p) Construct right linear grammar and regular expression corresponding to FA



$S = q_1$
$A = q_2$
$B = q_3$

RG: $S' \to 0A \mid \varepsilon$
$S \to 0A$
$A \to 1B$
$B \to 0A \mid 1B \mid 0S \mid 0$

$$\begin{cases} q_1 = 0 \cdot q_2 + \varepsilon \\ q_2 = 1 \cdot q_3 \\ q_3 = 1 \cdot q_3 + 0 \cdot q_2 + 0 \cdot q_1 \end{cases}$$

$\Rightarrow q_3 = 1 q_3 + 0(1 q_3) + 0 \cdot q_1$
$= 1 q_3 + 01 q_3 + 0 q_1$
$= (1+01) q_3 + 0 q_1$

$X = AX + B$
$\Rightarrow X = A^* B$ (Lema lui Arden)

$\Rightarrow q_3 = (1+01)^* 0 q_1$

$q_2 = 1 q_3 \Rightarrow q_2 = 1(1+01)^* 0 q_1$

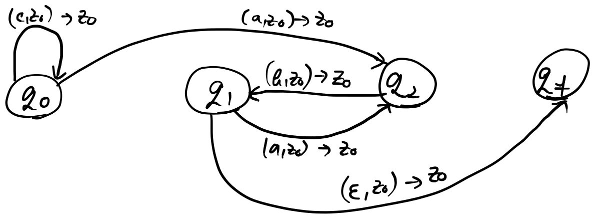
$q_1 = 0 q_2 + \varepsilon$
$= 01(1+01)^* 0 q_1 + \varepsilon$

$\Rightarrow \boxed{q_1 = (01(1+01)^* 0)^*}$

III. (2p) Construct a push-down automaton corresponding to the language: $L = \{c^m (ab)^n \mid n>0, m>=0\}$

$\delta(q_0, c, Z_0) = (q_0, Z_0)$
$\delta(q_0, a, Z_0) = (q_2, Z_0)$
$\delta(q_2, b, Z_0) = (q_1, Z_0)$
$\delta(q_1, a, Z_0) = (q_2, Z_0)$
$\delta(q_1, \varepsilon, Z_0) = (q_f, Z_0)$

$M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$
$Q = \{q_0, q_1, q_2, q_f\}$
$\Sigma = \{a, b, c\}$
$\Gamma = \{Z_0\}$
$F = \{q_f\}$



$\not\$ \ \varepsilon$

	a	b	c	d	\$
S	AB, 1	AB, 1	AB, 1		
A	a, 3	bA, 4	ε, 2		
B			cC, 5		
C				d, 7	ε, 6
a	pop				
b		pop			
c			pop		
d				pop	
\$					acc

IV.(2p) Consider the grammar G = ({S,A,B,C},{a,b,c,d},P,S) where

P: S -> AB [1]
A -> ε [2] | a [3] | bA [4]
B -> cC [5]
C -> ε [6] | d [7]

	FIRST	FOLLOW
S	a,b,c	ε
A	a,b,ε	c
B	c	ε
C	ε,d	$\not\$$ ε

Construct LL(1) table and parse the sequence w = bbac.

$w = bbac \in L(G)$?

$(bbac\$, S\$, \varepsilon) \vdash_{push} (bbac\$, AB\$, 1)$
$\vdash_{push} (bbac\$, bAB\$, 14) \vdash_{pop}$
$(bac\$, AB\$, 14) \vdash_{push} (bac\$, bAB\$, 144)$
$\vdash_{pop} (ac\$, AB\$, 144) \vdash_{push}$
$(ac\$, aB\$, 1443) \vdash_{pop} (c\$, B\$, 1443)$
$\vdash_{push} (c\$, cC\$, 14435) \vdash_{pop}$
$(\$, C\$, 14435) \vdash_{push} (\$, \$, 144356)$ ✓

P: S -> AB
A -> ε | a | bA
B -> cC
C -> ε|d

FIRST

	F_0	F_1	F_2
S	∅	{a,b,c}	{a,b,c}
A	{ε,a,b}	{ε,a,b}	{ε,a,b}
B	{c}	{c}	{c}
C	{ε,d}	{ε,d}	{ε,d}

FOLLOW:

	F_0	F_1	F_2	F_3
S	{ε}	{ε}	{ε}	{ε}
A	∅	{c}	{c}	{c}
B	∅	{ε}	{ε}	{ε}
C	∅	∅	{ε}	{ε}

S -> Bb | Cd

FIRST:

$B \to aB \mid \varepsilon$

$C \to cC \mid \varepsilon$

	F_0	F_1	F_2
S	$\emptyset$	{a,b,c,d}	{a,b,c,d}
B	{a, ε}	{a, ε}	{a, ε}
C	{c, ε}	{c, ε}	{c, ε}

FOLLOW:

	F_0	F_1	F_2	F_3
S	{ε}	{ε}	{ε}	
B	$\emptyset$	{b}	{b}	
C	$\emptyset$	{d}	{d}	

Construct LL(1) table and parse the sequence ...

V. (2p) Define a grammar corresponding to roman numbers with 2 digits (see table). Then, define an attribute grammar to computer number in base 10.

value	
1	I
5	V
10	X
50	L
100	C
500	D

IV →
VI

$S \to R_1 R_2$

$R \to I \mid V \mid X \mid L \mid C \mid D$

Production Rule	Semantic Rule

$S \to R_1 R_2$	if (R_1.val < R_2.val)
	S.val = R_2.val − R_1.val
	else
	S.val = R_1.val + R_2.val

$R \to I$	R.val = 1
$R \to V$	R.val = 5
$R \to X$	R val = 10
$R \to L$	R.val = 50
$R \to C$	R.val = 100
$R \to D$	R.val = 500

V.(2p) Define a grammar corresponding to arithmetical expressions (with operators for addition and substraction) and then define an attribute grammar for counting number of operators and number of operands.

ex :

$S \to S + T$

$S \to S - T$

$S \to T$

$T \to num$

Let ops the attribute for counting the number of operators.

Let opnds the attribute for counting the number of operands

Production Rule	Semantic Rule
$S \to S + T$	S.ops = S.ops + T.ops + 1 S.opnds = S.opnds + T.opnds
$S \to S - T$	S.ops = S.ops + T.ops + 1 S.opnds = S.opnds + T.opnds
$S \to T$	S.ops = T.ops S.opnds = T.opnds
$T \to num$	T.ops = 0 T.opnds = 1.