BITS, Pilani- K K Birla Goa Campus

Course: CS F363 Compiler construction | **Semester**-II, 2022-23, **MidSem Marks: 70** | **Weightage:** 35% **Mode:** Open Book | Date: 18/03/2023 | Time: 11:00-12:30 | Duration: 90 min

1. Consider the regular expression $r = aa(a|b)c^*(d|e)^*$.
   (a) Show a step by step construction of an NFA for the regular expression using Thompson construction method. Create DFAs from the basis. Next, we are expecting 5 NFAs for 5 important sub-expressions, constructed mechanically using inductive steps, without skipping any step. Next, show the final NFA. You MUST name the states as $0, 1, 2, 3, ...$in the final NFA. **6**
   (b) Assign numbers starting from 0 to each state you create right from the basis. What is the last number after completing 5 subexpressions? Explain why it is $2|r|$? **4**

2. Answer the following:
   (a) Write a set of production rules that generates the language $L(G) = \{a^n c b^m | n, m \geq 0\}$. Use the start symbol to be S. From these rules, explain the type of the grammar. **4**
   (b) For the set $L(G) = \{a^m, b^n, c^p, d^q | m, n, p, q \geq 1\}$, write a set of production rules. Using these rules, explain why it is regular. Draw its DFA. **8**
   (c) Now if $m == n$ and $p == q$, what will be the type of this grammar? Write production rules for this grammar $R(G) = \{a^m, b^m, c^p, d^p | m, p \geq 1\})$. **4**
   (d) Consider two languages $L_1 = \{a^n b^n c^m | n, m > 0\}$ and $L_2 = \{a^n b^m c^m | n, m > 0\}$. Write production rules for $L(1)$ and $L(2)$ and comment on their language types. Explain what is the type of the language $L_3 = L_1 \cap L_2$? **4**

3. Consider the production rules P=$\{1.E \to E+T$, $2.E \to T$, $3.T \to TF$, $4.T \to F$, $5.F \to F*$, $6.F \to a$, $7.F \to b\}$ for the grammar $G = \langle \{E, T, F\}, \{\mathtt{a}, \mathtt{b}, +, *\}, P, E \rangle$
   (a) Compute the FIRST and FOLLOW sets of all NTs. **9**
   (b) Compute the set of LR(0) items $I_0, I_1, \cdots$ **5**
   (c) Construct the SLR parsing table **10**
   (d) Identify the states where the shift and reduce conflicts have been resolved during SLR parsing table construction. **3**
   (e) Show every step of the SLR parser movements for the string `a+ab*` strictly in the following manner:

Stack Content	Action taken
\$(I0)	shift ?
\$(I0)a(I?)	R(6) $F \to a$
\$(I0)F(I?)	R(4) $T \to F$
.....	.....
\$(I0)??	accept

   The first few steps have been shown above, as a guidance that you must follow. **7**

4. Consider the following production rules of a grammar $A \to \alpha|\beta$. Let us define a new function $FIRST^+(A \to \alpha) = FIRST(\alpha) \cup FOLLOW(A)$ if $\alpha \Rightarrow^* \epsilon$, otherwise $FIRST^+(A \to \alpha) = FIRST(\alpha)$.

   Prove that for both the cases `(i)` neither $\alpha, \beta \not\Rightarrow^* \epsilon$ and `(ii)` only one of α or β, $\Rightarrow^* \epsilon$
   IF $(FIRST(\alpha) \cap FIRST(\beta) = \emptyset)$ THEN, $FIRST(\beta) \cap FOLLOW(A) = \emptyset$ implies $FIRST^+(A \to \alpha) \cap FIRST^+(A \to \beta) = \emptyset$ and vice versa. **6**

END