Formal Languages and Automata Theory **Name:** _____
SS2020 **Group:** _____
Exam, Variant Year 3
June 16th, 2020
Time Limit: 60 Minutes

This exam contains 5 questions. Total of points is 10.

Grade Table (for teacher use only)

Question	Points	Score
1	2	
2	2	
3	2	
4	2	
5	2	
Total:	10	

Remarks:

- All subjects are mandatory.
- Carefully read and apply the instructions from the Exam assignment on Google Classroom.
- Choose none, one of more of the variants for the problems below. You have to explain why you chose that answer and why you did not the others.

1. (2 points) For a mobile application you are coding, you have to store in the database only the Romanian phone numbers (e.g. +40724370012 / 0040724370012) and email addresses with a Romanian domain (e.g. ion@domeniu.ro; popescu.maria@dom.ro). The phone numbers must have exactly 9 digits after the Romanian prefix 0040/+40. The name of the domain and the extension of the email are written in lowercase letters and are case sensitive. Which of the following regular expressions express your needs:
   1. $(+407|00407)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)$;
      $((a|...|z|A|...|Z)^+.(a|...|z|A|...|Z)^+@(a|...|z)^+.ro)\,|((a|...|z|A|...|Z)^+@(a|...|z)^*.ro)$
   2. $(+407|00407)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)$;
      $((a|...|z|A|...|Z)^+.(a|...|z|A|...|Z)^+@(a|...|z)^+.ro)\,|\,((a|...|z|A|...|Z)^+@(a|...|z)^+.ro)$
   3. $(+407|00407)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)$;
      $((a|...|z|A|...|Z)^+.(a|...|z|A|...|Z)^+@(a|...|z)^+.ro)\,((a|...|z|A|...|Z)^+@(a|...|z)^+.ro)$
   4. $(+407|00407)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)(0|...|9)$;
      $((a|...|z|A|...|Z)^+.(a|...|z|A|...|Z)^+@(a|...|z)^+.ro)\,|\,((a|...|z|A|...|Z)^+@(a|...|z)^+.ro)$
2. (2 points) Assume you have to build push-down automaton simulating a natural language processor recognizing numerical palindromes. Which of the following grammars $G = (V_N, V_T, S, P)$ help you building the solution:
   1. $V_N = \{S\}, V_T = \{0, ..., 9\}, S, P = \{S \to 0S0|1S1|...|9S9|\lambda\}$
   2. $V_N = \{S\}, V_T = \{0, ..., 9\}, S, P = \{S \to 0S0|1S1|...|9S9|0|...|9\}$
   3. $V_N = \{S\}, V_T = \{0, ..., 9\}, S, P = \{S \to 0S0|1S1|...|9S9|0|...|9|\lambda\}$

3. (2 points) Assume you have a program, P, written in C, recognizing the language of even binary numbers. You would like to find an optimal program, w.r.t. memory and storage consumption, which does the same job. How would you proceed:
   1. use pointers instead of arrays and eliminate unused variables;
   2. rewrite the program such that it implements the minimal deterministic finite automaton recognizing the above mentioned language;
   3. rewrite the program such that it implements the nondeterministic finite automaton which recognizes the same language.

4. (2 points) Assume you have to construct a finite automata recognizing the keywords FOR and FORK. Which grammar $G = \{V_N, V_T, S, P\}$ would you use for that purpose:
   1. $V_N = \{S\}$, $V_T = \{F, O, R, K\}$, S, $P = \{S \to FOR|FORK\}$
   2. $V_N = \{S\}$, $V_T = \{FOR, FORK\}$, S, $P = \{S \to FOR|FORK\}$
   3. $V_N = \{S, x_1, x_2, x_3\}$, $V_T = \{F, O, R, K\}$, S, $P = \{S \to Fx_1, x_1 \to Ox_2, x_2 \to Rx_3|R, x_3 \to K\}$

5. (2 points) Consider the set of all strings of balanced parentheses of two types: round and square. An example of where these strings come from is as follows. If we take expressions in C, which use round parentheses for grouping and for arguments of function calls, and use square brackets for array indexes, and drop out everything but the parentheses, we get all strings of balanced parentheses of these two types. For example, $f(a[i] * (b[i][j], c[g(x)]), d[i])$ becomes the balanced parentheses string $([](][][()])[])$. A grammar $G = \{V_N, V_T, S, P\}$ for generating the strings of round and squared parentheses that are balanced is:
   1. $V_N = \{S\}$, $V_T = \{(,), [,]\}$, S, $P = \{S \to SS|[S]|(S)|()|[]\}$
   2. $V_N = \{S\}$, $V_T = \{(,), [,]\}$, S, $P = \{S \to (S)|[S]|()|[]|\lambda\}$
   3. $V_N = \{S\}$, $V_T = \{(,), [,]\}$, S, $P = \{S \to SS|(S)|[S]|\lambda\}$