TCS II
Formal Languages and Computability 2018/19
2nd Midterm (A)

29. May 2019

Solve the assignments on your own.

Time limit is 90 minutes.

Good luck!

ASSIGNMENT	POINTS	OUT OF	ASSIGNMENT	POINTS	OUT OF
1			2		
3			4		

FIRST AND LAST NAME: ⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯

STUDENT ID: ⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯

SIGNITURE: ⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯

1. Assignment: (25 points)

Let's define the language:

$$L_1 = \left\{ a^n b^{2n} \mid n > 0 \right\}$$

QUESTIONS:

1. Construct a PDA for L_1. For the PDA, write down the 7-tuple defining it!

2. Write the grammar for L_1.

2. Assignment: (25 points)

You are given the following context free grammar (CFG) G, $\Sigma = \{a, b, c, d\}$:

$S \rightarrow ACBD$
$A \rightarrow aA \mid a$
$B \rightarrow b \mid A$
$C \rightarrow c$
$D \rightarrow dD \mid dd \mid a$

QUESTIONS:

1. Turn this grammar into Chomsky Normal Form (CNF) (if needed).

2. Using the CYK algorithm, check if the word $acadd$ is in the language defined by grammar G.
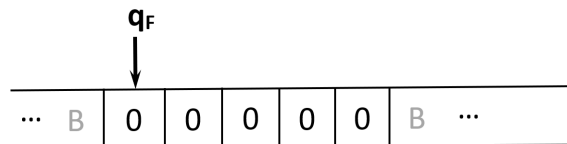
3. Assignment: (30 points)

Construct a Turing machine (TM) that performs the addition of 2 natural numbers (greater than 0) in "unary" notation. Each of the numbers must be represented as a sequence of all 0s; the two numbers should be separated by a single 1; B denotes the blank tape. Start in state q_0 with the read/write head of the TM positioned at the leftmost character representing the first number.
This picture represents the initial configuration of your TM with the numbers 2 and 3 on the input tape:

q_0

| ··· | B | 0 | 0 | 1 | 0 | 0 | 0 | B | ··· |
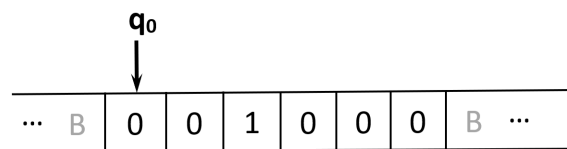
The result of your TM should be a single natural number, also in "unary" notation, with the read/write head positioned at its leftmost character, and the control unit in the (only) final state q_F.
This picture represents the final configuration of your TM with the results 5 $(= 2 + 3)$ on the input tape:

q_F

| ··· | B | 0 | 0 | 0 | 0 | 0 | B | ··· |

QUESTIONS:

1. Write down the δ-function of the TM.

2. Using the Instantaneous descriptions (IDs), show how your TM computes $2 + 3 = 5$.

4. Assignment: (20 points)

You are given the following Boolean (or logical) expression (X, Y and Z are boolean variables; $concatenation$, $+$ and $-$ represent the operations AND, OR and NOT, respectively):

$$(X + Y)(-Z)$$

QUESTIONS:

1. Convert the given Boolean expression first to Conjunctive Normal Form (CNF) (if needed) and then to 3–Conjunctive Normal Form (3–CNF).

2. What is the budget of the Vertex Cover (VC) generated from the 3–CNF in the previous question?

   HINT: You do not have to actually make the whole conversion from 3–CNF to VC; just write down the budget.