COMPUTER ENGINEERING DEPARTMENT

SUBJECT: THEORY OF COMPUTER SCIENCE

COURSE: T.E. **Year: 2020-2021** **Semester: V**

DEPT: Computer Engineering

SUBJECT CODE: CSC504 **EXAMINATION DATE: 14/01/2021**

THEORY OF COMPUTER SCIENCE ANSWER SHEET

Name : AMEY MAHENDRA THAKUR

Seat No.: 51112146

Exam : SEMESTER V

Subject : THEORY OF COMPUTER SCIENCE

Date : 14/01/2021

Day : THURSDAY

Student Signature: Amey

Q.3 A) i)

Post Correspondance Problem (PCP)

— The Post Correspondance Problem (PCP) introduced by Emil Post in 1946, is an undecidable decision problem.

Definition

Let A and B be two non empty lists of strings over Σ
A and B are given as below:

$A = \{x_1, x_2, x_3, \ldots x_k\}$

$B = \{y_1, y_2, y_3, \ldots y_k\}$

There is a post correspondance between A and B if there is a sequence of one or more integers $i, j, k, \ldots m$ such that:

The string $x_i, x_j, \ldots x_m = y_i, y_j, \ldots y_m$

Example

Does the PCP with two lists:

$A = \{a, aba^3, ab\}$ and

$B = \{a^3, ab, b\}$.

Have a solution?

→ So to find a sequence using which when the elements of A and B are listed, will produce identical strings

The required sequence is (2, 1, 1, 3)

$A_2 A_1 A_1 A_3 = aba^3 \; a \; a \; ab = ab \, a^6 b$

$B_2 B_1 B_1 B_3 = aba^3 \; a^3 \; b = ab \, a^6 \, b$

Thus, the PCP has the solution

So accept the undecidability of post correspondance Problem without proof.

14-01-2021

Example:

Determining the solution for the following instance of PCP

	List A	List B
i	w_i	x_i
1	01	0
2	110010	0
3	1	1111
4	11	01

The PCP has a solution.

The required sequence is (1, 3, 2, 4, 4, 3)

$w_1 w_3 w_2 w_4 w_4 w_3$ = 01111 0010 11111
$x_1 x_3 x_2 x_4 x_4 x_3$ = 01111 0010 11111

Q.3. A) iii)

Universal Turing Machine

- A Turing Machine is said to be universal TM if it can accept:
  1. The input data
  2. An algorithm (description) for computing.
- This is precisely what a general purpose digital computer does. A digital computer accepts a program written in high level language. Thus, a general purpose Turing machine will be called a Universal Turing machine if it is powerful enough to simulate the behaviour of any digital computer including any Turing machine itself.
- More precisely, a Universal turing machine can simulate the behaviour of an arbitrary Turing machine over any set of input symbols. Thus it is possible to create a single machine that can be used to compute any computable sequence.
  If this machine is supposed to be supplied with the tape on the beginning on which is written the input string of quintuple separated with some special symbol of some computing machine M. then the universal Turing Machine U will compute the same strings as those by M.
- The model of universal Turing machine is considered to be a theoretical breakthrough that led to the concept of stored programmer computing device.

- Designing a general purpose Turing machine is a more complex task. Once the transition of Turing Machine is defined, the machine is restricted to carrying out one particular type of computation.
- Digital computers, on the other hands, are general purpose machines that cannot be considered equivalent to general purpose digital computers until they are designed to be reprogrammed.
- By modifying our basic model of a Turing Machine we can design a universal turing machine. The modifred Turing Machine must have a large number of states for stimulating even a simple behaviour. We modify our basic model by:
  1. Increase the number of read/write heads
  2. Increase the number of dimensions of input tape
  3. Adding a special purpose memory
- All the above modifications in the basic model of a Turing Machine will almost speed up the operations of the machine can do.
- A number of ways can be used to explain to show that Turing machines are useful models of real computers. Anything that can be computed by a real computer can also be computed by a Turing Machine. A Turing Machine, for example can simulate any type of functions used in programming languages. Recursion and parameters passing are some typical examples. A Turing machine can also be used to simplify the statements of an algorithm.

- A Turing machine is not very capable of handling it in a given finite amount of time.
  Also, Turing machines ~~are~~ are not designed to receive unbounded input as many real programmers like word processors, operating systems and other software systems.

Q.3 B. ii)

Design PDA for $L = \{a^{2n}b^n, n \geq 1\}$

① Logic : For each two 'a', push one 'x' into Stack.
For each 'b', pop one 'x' from Stack.

② Machine Definition :
Let $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ be the required PDA.

$Q = \{q_0, q_1, q_2, q_f\}$
$\Sigma = \{a, b\}$
$\Gamma = \{x, Z_0\}$
Z_0 = Stack Top
q_0 = Initial state
$F = \{q_f\}$

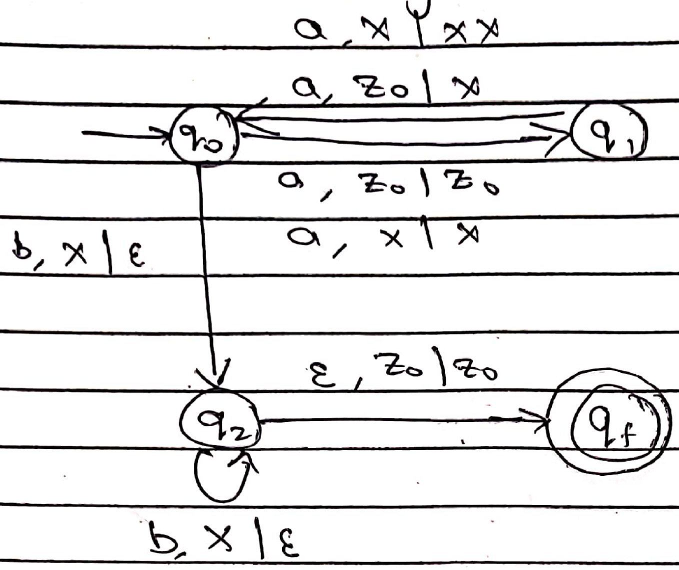
③ Rules:

$\delta(q_0, a, Z_0) = \{(q_1, Z_0)\}$ here for 1st 'a' bypass
$\delta(q_1, a, Z_0) = \{(q_0, x)\}$
$\delta(q_0, a, x) = \{(q_1, x)\}$
$\delta(q_1, a, x) = \{(q_0, xx)\}$
$\delta(q_0, b, x) = \{(q_2, \varepsilon)\}$
$\delta(q_2, b, x) = \{(q_2, \varepsilon)\}$
$\delta(q_2, \varepsilon, Z_0) = \{(q_f, Z_0)\}$

Name: AMEY MAHENDRA THAKUR Branch: COMPUTER Seat No.: 51112146

Subject: THEORY OF COMPUTER SCIENCE Exam: SEMESTER V Page No.: 7/7

(4) Transition Diagram:



(5) Simulation:

Consider the string "aaaabb"

$\delta(q_0, aaaabb, z_0)$

$\rightarrow \delta(q_1, aaabb, z_0)$

$\rightarrow \delta(q_0, aabb, Xz_0)$

$\rightarrow \delta(q_1, abb, Xz_0)$

$\rightarrow \delta(q_0, bb, XXz_0)$

$\rightarrow \delta(q_2, b, Xz_0)$

$\rightarrow \delta(q_2, \varepsilon, z_0)$

$\rightarrow \delta(q_f, z_0)$ Accepted

14-01-2021 **Student Signature:** Amey