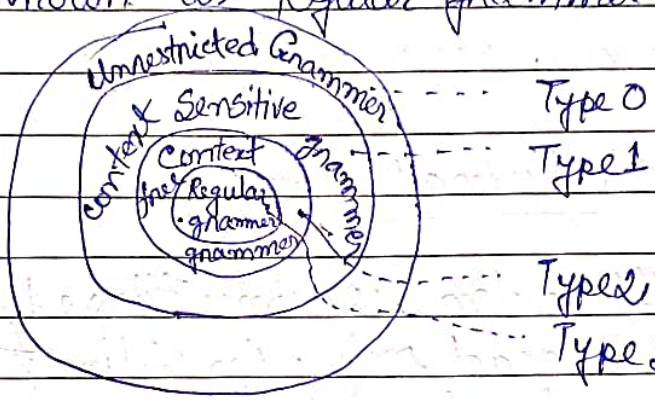
TOC Test-3

Name: Zishmendu Sarker
Roll: 2k19/CO/450
Date .....

Answer to the ques. no: 2

Chomsky hierarchy represents the class of languages that are accepted by the different machine. According to chomsky hierarchy, grammer is divided into 4 types.

① Type 0 known as unrestricted grammar
② Type 1 known as Context sensitive grammer.
③ Type 2 known as Context Free grammer.
④ Type 3 known as Regular grammer.



There is a hierarchy. Therefore every language of type 3 is also of type 2, 1, 0. Similarly every language of type 2 is also type 1, 0, and rest are like this sequence.

Type 0 grammer:

Type 0 grammers generate recursively enumerable languages. The productions have no restrictions. They are any phase structure grammers including all formal grammers. They generate the language that are recognised by a turing machine. The production can be in the form of $\alpha \to \beta$ where α is a string of terminals and non terminals with at least one non-terminal and α cannot be null. β is a

Date .....

string of terminals and non-terminals.

Ex.- $S \rightarrow ACaB$
$Bc \rightarrow acB$
$CB \rightarrow DB$
$aD \rightarrow Db$

Type-1 Grammar :

It generates context sensitive languages. The production must be in the form of

$$\alpha A \beta \rightarrow \alpha \gamma \beta$$

where $A \in N$ (non-terminal)
and $\alpha, \beta, \gamma \in (T \cup N)^*$ (String of terminals and non-terminals)

The string α and β may be empty, but γ must be non-empty. The rule $S \rightarrow \varepsilon$ allowed if S does not on the right side of any rule.

Ex. $AB \rightarrow AbBc$
$A \rightarrow bcA$
$B \rightarrow b$

Type-2 Grammars :

It generates context free ~~grammar~~ languages. The productions must be in form $A \rightarrow \gamma$.

where $A \in N$ (non-terminal)
and $\gamma \in (T \cup N)^*$ (string of terminals and non-terminals)

The language generates by these grammers are be recognised by a non-deterministic pushdown Automata

Spiral

Date .....

Ex. $S \to Xa$
$X \to a$
$X \to aX$
$X \to abc$
$X \to \varepsilon$

Type-3 Grammars:

It generates regular languages. It must have a single non-terminal on the left-hand side consisting of a single terminal followed by a single non-terminal. The production must be in the form $X \to a$ or $X \to aY$. where $X, Y \in N$ (Non-terminals) and $a \in T$ (Terminal)
The rule $S \to \varepsilon$ doesnot appear on the side of any rule.
Ex.
$X \to \varepsilon$
$X \to a \mid aY$
$Y \to b$.

—

Answer to the ques. no: 3

A turing machine (TM) is a mathematical model which consists of an infinite length tape divided into cells on which input is given. It consists of a head which reads the input tape. A state register stores the state of the turing machine. After reading an input symbol, it is replaced by another symbol. Its internal state is changed.

Date .....

and it moves from one cell to the right or left. If the M reaches the final state, the input string is accepted, otherwise not rejected.

A turing machine can be formally described as 7 tuples

$(Q, X, \Sigma, \delta, q_0, B, F)$

where,

Q = is a finite set of states
X = the tape Alphabet
Σ = input Alphabet
δ = the transition ; $\delta : Q \times X \rightarrow Q \times X \times \{$ left-shift, right-shift $\}$
q_0 = initial state.
B = the black symbol
F = the set of infinite state.

For a turing machine, the time complexity refers to the measure of the number of times the tape moves when the machine is initialized of some input symbol and the space complexity is the number of cells of the tape written.

Time complexity all reasonable function :-

$$T(n) = O(n \log n)$$

Space complexity

$$S(n) = O(n)$$

Date .....

Ex.

For TM

$Q = \{q_0, q_1, q_2, q_n\}$

$X = \{a, b\}$

$\Sigma = \{1\}$

$q_0 = \{q_0\}$

B = black symbol

$F = \{q_F\}$.

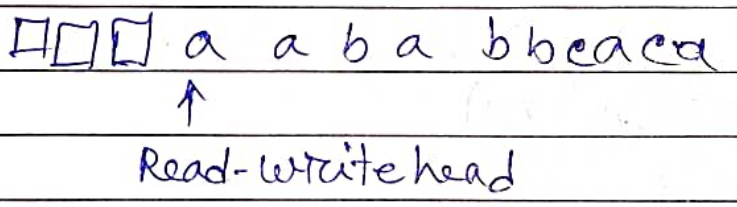
Present State	Present State	Present State
q_0	q_1	q_2
$1 R q_1$	$1 L q_0$	$1 L q_4$
$1 L R q_{f_2}$	$1 R q_1$	$1 R q_2$

Answer to the ques. no: 4

(a) The variations of turing machine are as follows:-

(1) Turing machine with two- sided infinite tapes :

The first variable we consider is that of having a turing machine with a tape that is infinite on both sides.



Double- sided infinite taps can express any turing machine with a single- ~~tape~~ side infinite tape by leaving the tm definition unaltered and simply not using the negative index side.

of the tape.

(ii) Turing machine with K- work-tapes :-

Considering TMs are fixed, but ~~arbitary~~ arbitary number K of tapes, these variable also have K independent heads and typically one tape as input/output tape and use the remaining K-1 tapes as a scratch pad to record and compute on intermediate computation that are related to the input and output functions.

$$\delta_K : Q \times \Sigma^K \rightarrow (Q \cup \{q_H\}) (\Sigma \cup \{L, R\})^K$$

(iii) Non-deterministic turing machine :- (NDTM)

NDTM are normal turing machines with the only difference being that the transition function may give us a number of possibilities that we can chose for next step to take.

$$\delta : Q \times \Sigma \rightarrow P((Q \cup \{q_H\}) \times (\Sigma \cup \{L, R\})).$$

The state of the variation of the standard Model of TM :-

- → Stay-option
- → Semi-Infinite
- → OFF-line
- → Multitape
- → Multidimentional
- → Non-deterministic.

TOC Test-3

Answer to the ques. no: 1

Multitape:- multi-tape tm constructed with multiple tapes where each is designed with a unique head. Here, each head independently with other heads. While at the starting input is noted on tape 1 and others are noted empty. Initially the first tape includes an input and other tapes are remained empty.

Offline multiple Turing Machine:-

An offline tm is a multitape tm where input tape is read only. An offline tm can simulate any tm by using one more tape than tm. A tm with several tapes is said to be multiple tm. In a multiple tm, each step is controlled its own independent read/write head.