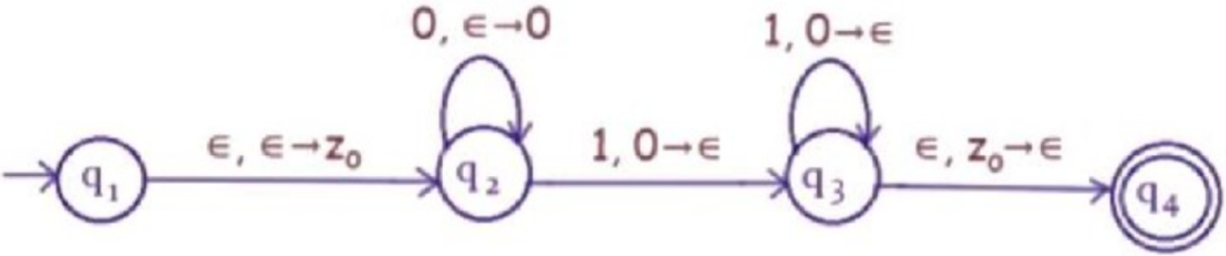
The following PDA:



can be used to represent the following language:

- ( ) ww^R
- (•) 0^n1^n, where n>=0
- ( ) The language that accepts any number of 0's followed by any number of 1's: 0^*1^*
- ( ) $(0^n1^n)^R$

A language is said to be context-free if:

- [ ] There exist deterministic-finite automata that can represent the language
- [x] There exists a context-free grammar that accept the language
- [ ] There exists a regular left linear grammar that accepts the language
- [ ] There exists a turing machine with multi-tape that can represent the language

Which of the following grammar is in BNF form:

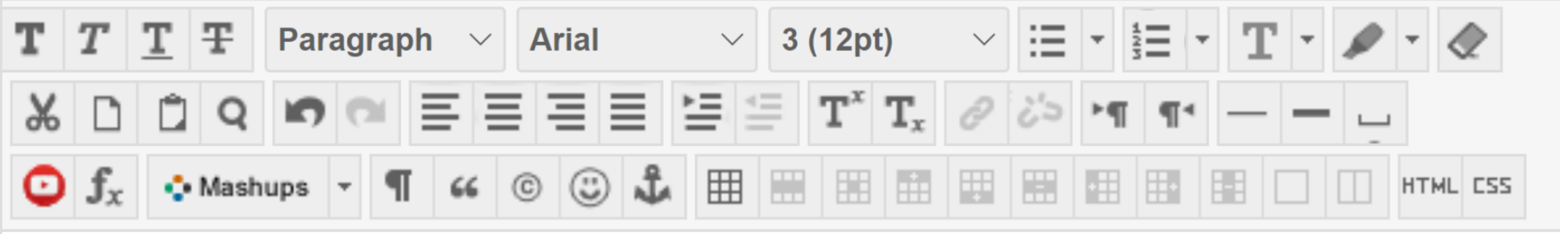
- [ ] $S -> aS \mid Sa \mid \epsilon$
- [x] $S ::= a<S> \mid <S>a \mid a$
- [ ] $S ::= aS \mid Sa \mid \epsilon$
- [ ] $S -> a<S> \mid <S>a \mid \epsilon$

We can classify languages into several sets based on the type of words accepted. However, the largest set is:

- [ ] Context-Free Languages
- [ ] Regular Languages
- [ ] non-regular languages
- [x] languages accepted by turing machines that is called enumerable recursive languages

Write a grammar that accepts the language: $L = wccw^R$

For the toolbar, press ALT+F10 (PC) or ALT+FN+F10 (Mac).



Push-Down automata is used to represent the following class of languages:

- (selected) context-free languages
- regular languages but not CFL
- recursively enumerable languages
- Languages that can be represented using turing machines

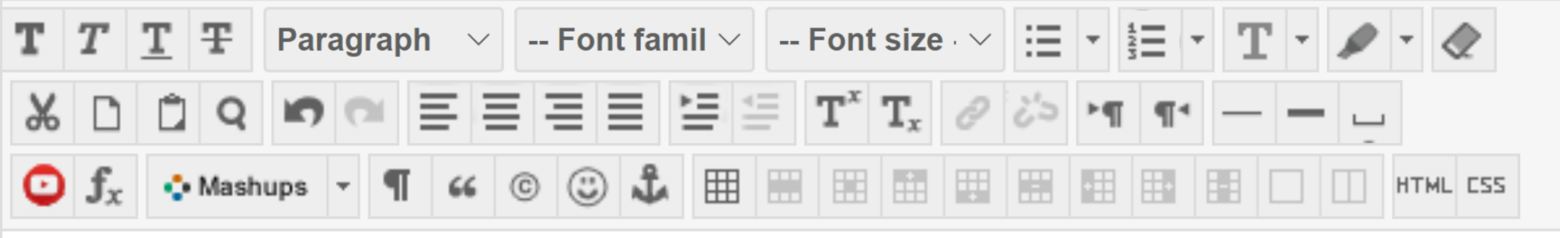
Using the derivation form:

example: SAB=>SAb=>Sab=>ab

Please show whether this grammar is ambiguous or not:

$$S -> aS \;|Sa|\; \epsilon$$

For the toolbar, press ALT+F10 (PC) or ALT+FN+F10 (Mac).



Using the pumping lemma, we can divide the string into the following string:

- [ ] $x^n yz$, where $|xy| > m$
- [x] $xy^n z$, where $|xy| <= m$
- [ ] xyz^n, where $y|z| <= m$
- [ ] $xy^n z$, where $|xy| = |yz|$

We can represent transitions in push-down automata as the following:

$$\delta(q_1, a, 1) = \{(q_1, 11)\},$$

We can say the following about this transition:

- ○ We are in state q1, when we read an 'a' we pop 1 then switch to state q1 and then push '1' to the stack.
- ○ We are in state q1, when we read an 'a' we pop 1 then switch to state q2 and then push '11' to the stack.
- ◉ We are in state q1, when we read an 'a' we pop 1 then switch to state q1 and then push '11' to the stack.
- ○ We are in state q1, when we read an '1' we pop 'a' then switch to state q1 and then push 'aa' to the stack.

Given the following definition of PDA:

Q is a finite set of internal states of the control unit,

Σ is ###########,

Γ is a ##########,

δ : Q × (Σ ∪ {λ}) × Γ → set of finite subsets of Q × Γ is the transition function, q0 ∈ Q is the initial state of the control unit,*

z ∈ Γ is ########## ,

F ⊆ Q ###########

Please state what is *z*

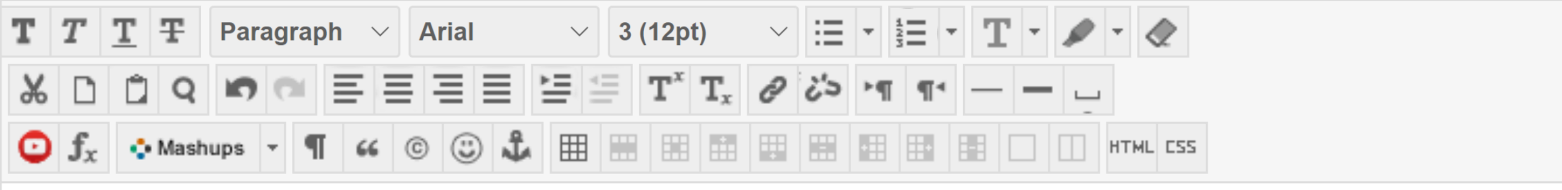
- ○ The set of alphabet for the stack
- ◉ The starting symbol of the stack
- ○ The set of final states in the PDA
- ○ it indicates the iitial state of the PDA

We use the pumping the lemma to prove that:

- [ ] a Language is context-free languages
- [ ] a L is a regular language
- [x] a language is not a regular language
- [ ] Is recursively enumerable

What is the difference between NPDA and DPDA. (non-deterministic pushdown automata and deterministic pushdown automata)

For the toolbar, press ALT+F10 (PC) or ALT+FN+F10 (Mac).



A grammar is said to be context-free is all its productions are of the type:

- [ ] $A \rightarrow x$, where $A \in V$ and $x \,!\!\in (V \cup T)^*$
- [ ] $A \rightarrow xy^*z$, where $A \in V$ and $x,y \in (V \cup T)^*$ where $z \,!\!\in \Sigma$
- [x] $A \rightarrow x$, where $A \in V$ and $x \in (V \cup T)^*$
- [ ] $x \rightarrow A$, where $A \in V$ and $x \in (V \cup T)^*$

Draw a push-down automata that accept the language: $L = wcw^R$

Selected Answer:

Attach File

Browse My Computer

Which of the following grammar that represent the taibah university email address of the form: TU0987123@taibahu.edu.sa

- [ ] S::= "TU" <letter>* "@" <domain>
  <letter>::= [a−z]
  <domain>::= "taibahu.edu.sa"
- [ ] S::= "TU" <digit>* "@" <domain>
  <letter>::= [0−9]
  <domain>::= "taibahu.edu.sa"
- [x] S::= "TU" <digit>"@" <domain>,
  <digit>::= [0−9],
  <domain>::= "taibahu.edu.sa"
- [ ] S::= "TU" <digit>* "@" <domain>,
  <digit>::= [0−0],
  <domain>::= "taibahu.edu.sa"

Pumping Lemma is used by mathematicians to refer to the following observation:

- [x] If we put n>m objects in boxes, then at least 1 box will have one more object.
- [ ] If we put n<m objects in boxes, then at least 1 box will have one more object.
- [ ] Objects must be equal numbers as the boxes where m = n.
- [ ] To observe how the pigeon behave regarding which box they choose to go to.

The language L= ww^R can be classified as

- [x] context-free language and can be represnted using CFG
- [ ] Regular language and can be represented using regular grammar
- [ ] Context-free and regular language can be represented using NFA
- [ ] can be represented using DFA (or NFA)