Khuyen Le Thi Minh – s5128

Homework 11

1. *Build a pushdown automaton accepting bracket expressions with three kinds of brackets:* $()[]\{\}$, *assuming, that:*
   - *round brackets can onle be placed inside braces, square or round brackets,*
   - *square brackets can be placed only inside braces or square brackets,*
   - *braces can be put only inside braces.*

   For instance, $\{\{()[[()(())]]\}\}[]$ *is a correct expression, but* $[[\{\}]](\,[][])$ *is not.*

   $(s,x) \xrightarrow{(} (s,)x)\ for\ x = \perp,),], \}$

   $(s,x) \xrightarrow{[} (s,]x)\ for\ x = \perp,], \}$

   $(s,x) \xrightarrow{\{} (s,\}x)\ for\ x = \perp, \}$

   $(s,x) \xrightarrow{x} (s,\varepsilon)\ for\ x = \perp,),], \}$

   $(s,\perp) \xrightarrow{\varepsilon} (s,\varepsilon)$

   On the stack we push in the closing bracket if we encounter the opening bracket, with the rules that we first check the top of the stack for the symbol, and if that symbol satisfy our rule (if we push in a round bracket, at the top of the stack could be any bracket or pin, if we push in square bracket, the top of the stack cannot be rounded bracket, and if we push in a brace then at the top of the stack can be only a brace or a pin).
   If we encouter the closing bracket, we check if it's the one that appear on top of the stack, if yes then we remove it.

   Give a computation accepting $\{[()]()\}$.

   $(s,\perp) \xrightarrow{\{} (s,\}\perp)$ *First when { is in, we read the top of the stack is a pin, then we push } to stack,*

   $(s,\}) \xrightarrow{[} (s,]\})$ *then we have [in, we read top of the stack is } then we push] in,*

   $(s,]) \xrightarrow{(} (s,)])$ *then when (is in, after reading from top of stack, it's] we push) in,*

   $(s,)) \xrightarrow{)} (s,\varepsilon)$ *when we have) in, on top of the stack is), we remove it out from stack,*

   $(s,]) \xrightarrow{]} (s,\varepsilon)$ *when] is in, the top of stack is], we remove it,*

   $(s,\}) \xrightarrow{(} (s,)\})$ *when (is in, we read from top of stack is } we push) in,*

   $(s,)) \xrightarrow{)} (s,\varepsilon)$ *) is in, and the same bracket has been read from top of stack, we remove it,*

   $(s,\}) \xrightarrow{\}} (s,\varepsilon)$ *} is in and we read the same bracket from top of stack, we remove it,*

   $(s,\perp) \xrightarrow{\varepsilon} (s,\varepsilon)$ *the word end and empty symbol is in, we read top of stack is a pin and we remove it.*

2. *Convert the grammar:* $S \to aSa|bSb|a|b|\varepsilon$ *generating pallindromes (over an alphabet), into a pushdown automaton (using the algorithm shown in the lecture).*

   $(s,S) \xrightarrow{\varepsilon} (s,aSa)$

   $(s,S) \xrightarrow{\varepsilon} (s,bSb)$

   $(s,S) \xrightarrow{\varepsilon} (s,a)$

   $(s,S) \xrightarrow{\varepsilon} (s,b)$

   $(s,S) \xrightarrow{\varepsilon} (s,\varepsilon)$

   $(s,a) \xrightarrow{a} (s,\varepsilon)$

   $(s,b) \xrightarrow{b} (s,\varepsilon)$

3. *Convert the pushdown automaton accepting bracket expressions:*

   $(s,x) \xrightarrow{[} (s,]x) for\ x =], \perp$

   $(s,]) \xrightarrow{]} (s,\varepsilon)$

   $(s,\perp) \xrightarrow{\varepsilon} (s,\varepsilon)$

   into an equivalent context − free grammar (using the algorithm shown in the lecture).
   For clarity we'll transform our pushdown automation to the following form:

   $(s,\perp) \xrightarrow{[} (s,]\perp)$

   $(s,]) \xrightarrow{[} (s,]])$

   $(s,]) \xrightarrow{]} (s,\varepsilon)$

   $(s,\perp) \xrightarrow{\varepsilon} (s,\varepsilon)$

   So we'll have the grammar (replacing terminal] with nonterminal A):

   $S \to [AS|\varepsilon$

   $A \to [AA\,|\,]$

   Give a derivation for the word $[][[]]$.

   $S \to [AS \to [[AAS \to [[]AS \to [[][AS \to [[][[AAAS \to [[][[]AAS \to [[][[]]AS \to [[][[]]]S \to [][[]]$