Algorithm 4.6.1
The CYK algorithm

input: context-free grammar $G = (V, \Sigma, P, S)$
string $u = x_1 x_2 \dots x_n \in \Sigma^*$
private:
X : a table containing sets of variables
$step$: the index of the "diagonal", the main diagonal is 1, the one above it is 2, and so on.
i: row index (the column index is calculated from it)
k: split position in the string

// Initialize the entire table.
1. initialize all $X_{i,j}$ to $\emptyset$

// Initialize the main diagonal from the rules that derive the terminals of the string.
2. **for** $i = 1$ to n
for each variable A
if there is a rule $A \rightarrow x_{i,i}$ **then**
$X_{i,i} := X_{i,i} \cup \{A\}$

// Do for each "diagonal."
3. **for** $step = 2$ to n

// The cells start from $i, i + step - 1$.
3.1 **for** $i = 1$ to $n - step + 1$

// Do for each split position.
3.1.1 **for** $k = i$ to $i + step - 2$
if there are variables $B \in X_{i,k}, C \in X_{k+1,i+step-1}$, and a rule $A \rightarrow BC$ **then**
$X_{i,i+step-1} = X_{i,i+step-1} \cup \{A\}$

4. **if** $S \in X_{1,n}$ **then**
return TRUE
else
return FALSE