GABINETE, KEITH GINOEL S.   **CMSC 129 - CD-1L**

Exercise 4: Derivation

Instructions:

- Create a program that does derivation
- Inputs are the productions and the string to derive
- Output is the step by step derivation
- Input may be from a file or via command line
- Include instructions on how to compile and run your program (include sample file/input)
- Include screenshots of your program results

Assumptions:
- grammar is unambiguous
- input string is correct

Example:
Production
S -> S S + | S S * | a
String to derive: a a + a *

NOTE:

For the instructions on how to compile and run the program, refer to the '**README.md**' file included in the zip file.

PROGRAM RESULTS:

Valid inputs:

```
shio@X509FB:~/Desktop/repos/principles-of-compiler-design/04 - Derivation$ make
python3 GABINETEKG_exer4.py < input.txt
[NOTE:] If no valid derivation is found, either the input string cannot be derived from the given grammar, or the length of the current derivation has exceeded the limit tha
t this program can generate, which is 5 times the length of the input string.

[SUGGESTION: You may also try adjusting the limit 'TOLERANCE_MULTIPLIER' in the code to allow longer derivations if you are certain that your input can be derived using the
given set of production rules. However, if that doesn't work, the input might simply be too long for this program to process. It may take a long time, so you may choose to w
ait or terminate the program since the author is also unsure if the implementation is optimal due to using BFS to generate derivations.]
_____

String to derive: a a + a *

Leftmost derivation:
S
-> S S *
-> S S + S *
-> a S + S *
-> a a + S *
-> a a + a *

String to derive: a a *

Leftmost derivation:
S
-> S S *
-> a S *
-> a a *
```

```
shio@X509FB:~/Desktop/repos/principles-of-compiler-design/04 - Derivation$ make
python3 GABINETEKG_exer4.py < input.txt
[NOTE:] If no valid derivation is found, either the input string cannot be derived from the given grammar, or the length of the current derivation has exceeded the limit tha
t this program can generate, which is 5 times the length of the input string.

[SUGGESTION: You may also try adjusting the limit 'TOLERANCE_MULTIPLIER' in the code to allow longer derivations if you are certain that your input can be derived using the
given set of production rules. However, if that doesn't work, the input might simply be too long for this program to process. It may take a long time, so you may choose to w
ait or terminate the program since the author is also unsure if the implementation is optimal due to using BFS to generate derivations.]
_____

String to derive: n + n * n

Leftmost derivation:
A
-> C B
-> E D B
-> n D B
-> n # B
-> n # + C B
-> n # + E D B
-> n # + n D B
-> n # + n * E D B
-> n # + n * n D B
-> n # + n * n # B
-> n # + n * n # #
```

Invalid input:

```
shio@X509FB:~/Desktop/repos/principles-of-compiler-design/04 - Derivation$ make
python3 GABINETEKG_exer4.py < input.txt
[NOTE:] If no valid derivation is found, either the input string cannot be derived from the given grammar, or the length of the current derivation has exceeded the limit tha
t this program can generate, which is 5 times the length of the input string.

[SUGGESTION: You may also try adjusting the limit 'TOLERANCE_MULTIPLIER' in the code to allow longer derivations if you are certain that your input can be derived using the
given set of production rules. However, if that doesn't work, the input might simply be too long for this program to process. It may take a long time, so you may choose to w
ait or terminate the program since the author is also unsure if the implementation is optimal due to using BFS to generate derivations.]
_____

String to derive: n + n * n n

Leftmost derivation:
No valid derivation found!
```