

# example1-output

```
"C:\Program Files\Java\jdk1.8.0_181\bin\java" "-javaagent:D:\Software2\IntelliJ IDEA 2017.3.2\lib\idea_rt
.jar=58938:D:\Software2\IntelliJ IDEA 2017.3.2\bin" -Dfile.encoding=UTF-8 -classpath "C:\Program Files\
Java\jdk1.8.0_181\jre\lib\charsets.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\deploy.jar;C:\Program
Files\Java\jdk1.8.0_181\jre\lib\ext\access-bridge-64.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\ext\
cldrdata.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\ext\dnsns.jar;C:\Program Files\Java\jdk1.8.0_181\
jre\lib\ext\jaccess.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\ext\jfxrt.jar;C:\Program Files\Java\
jdk1.8.0_181\jre\lib\ext\localedata.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\ext\nashorn.jar;C:\
Program Files\Java\jdk1.8.0_181\jre\lib\ext\sunec.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\ext\
sunjce_provider.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\ext\sunmscapi.jar;C:\Program Files\Java\
jdk1.8.0_181\jre\lib\ext\sunpkcs11.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\ext\zipfs.jar;C:\
Program Files\Java\jdk1.8.0_181\jre\lib\javaws.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\jce.jar;C:\
Program Files\Java\jdk1.8.0_181\jre\lib\jfr.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\jfxswt.jar;C:\
Program Files\Java\jdk1.8.0_181\jre\lib\jsse.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\management-
agent.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\plugin.jar;C:\Program Files\Java\jdk1.8.0_181\jre\
lib\resources.jar;C:\Program Files\Java\jdk1.8.0_181\jre\lib\rt.jar;D:\GitOpenProject\Assignment\
compiler2018\compiler\out\production\LR1Grammar" Main
```

1.INPUT THE START NON\_TERMINAL:

>E

2.INPUT THE GRAMMAR:(copy E from there)

>E->E+(E)|i

>end

-----all productions

E:

[E+(E), i]

-----first

E:

[i]

-----DFA

Node 0:

□-> E [\$]

^

E-> E+(E) [\$ , +]

^

E-> i [\$ , +]

^

Node 1:

□-> E [\$]

^

E-> E+(E) [\$ , +]

^

Node 2:

E-> i [\$ , +]

^

Node 3:

E-> E+(E) [\$ , +]

^

Node 4:

E-> E+( E) [\$ , +]

^

E-> E+(E) [), +]

^

E-> i [), +]

^

Node 5:

E-> E+(E ) [\$ , +]

^

E-> E+(E) [), +]

^

Node 6:

E-> i [), +]

^

Node 7:

E-> E+(E) [\$ , +]

^

Node 8:

E-> E+( E) [), +]

^

Node 9:

E-> E+( E) [), +]

^

E-> E+(E) [), +]

^

E-> i [), +]

^

Node 10:

E-> E+(E ) [), +]

^

E-> E+(E) [), +]

^

Node 11:

E-> E+(E) [), +]

^

-----simplified DFA

node 0:

via E ==> 1

# example1-output

```

via i ==> 2
node 1:
  Reduce []->E on [$]
via + ==> 3
node 2:
  Reduce E->i on [$, +]
node 3:
  via ( ==> 4
node 4:
  via E ==> 5
  via i ==> 6
node 5:
  via ) ==> 7
  via + ==> 8
node 6:
  Reduce E->i on [], +]
node 7:
  Reduce E->E+(E) on [$, +]
node 8:
  via ( ==> 9
node 9:
  via E ==> 10
  via i ==> 6
node 10:
  via ) ==> 11
  via + ==> 8
node 11:
  Reduce E->E+(E) on [], +]

```

-----Parse Table

```

0 {E=go 1, i=shift 2}
1 {$=reduce []->E, +=shift 3}
2 {$=reduce E->i, +=reduce E->i}
3 {(=shift 4}
4 {E=go 5, i=shift 6}
5 {)=shift 7, +=shift 8}
6 {)=reduce E->i, +=reduce E->i}
7 {$=reduce E->E+(E), +=reduce E->E+(E)}
8 {(=shift 9}
9 {E=go 10, i=shift 6}
10 {)=shift 11, +=shift 8}
11 {)=reduce E->E+(E), +=reduce E->E+(E)}

```

3.INPUT THE SENTENCE:

>i+(i)+(i)

STATE_STACK	NOTATION_STACK	INPUT	ACTION
[0]	[\$]	i+(i)+(i)\$	shift 2
[0, 2]	[\$, i]	+(i)+(i)\$	reduce E->i
[0]	[\$, E]	+(i)+(i)\$	go 1
[0, 1]	[\$, E]	+(i)+(i)\$	shift 3
[0, 1, 3]	[\$, E, +]	(i)+(i)\$	shift 4
[0, 1, 3, 4]	[\$, E, +, (]	i)+(i)\$	shift 6
[0, 1, 3, 4, 6]	[\$, E, +, (, i]	)+(i)\$	reduce E->i
[0, 1, 3, 4]	[\$, E, +, (, E]	)+(i)\$	go 5
[0, 1, 3, 4, 5]	[\$, E, +, (, E]	)+(i)\$	shift 7
[0, 1, 3, 4, 5, 7]	[\$, E, +, (, E, )]	+(i)\$	reduce E->E+(E)
[0]	[\$, E]	+(i)\$	go 1
[0, 1]	[\$, E]	+(i)\$	shift 3
[0, 1, 3]	[\$, E, +]	(i)\$	shift 4
[0, 1, 3, 4]	[\$, E, +, (]	i)\$	shift 6
[0, 1, 3, 4, 6]	[\$, E, +, (, i]	)\$	reduce E->i
[0, 1, 3, 4]	[\$, E, +, (, E]	)\$	go 5
[0, 1, 3, 4, 5]	[\$, E, +, (, E]	)\$	shift 7
[0, 1, 3, 4, 5, 7]	[\$, E, +, (, E, )]	\$	reduce E->E+(E)
[0]	[\$, E]	\$	go 1
[0, 1]	[\$, E]	\$	reduce []->E
[0]	[\$, []]	\$	Accepted

Process finished with exit code 0