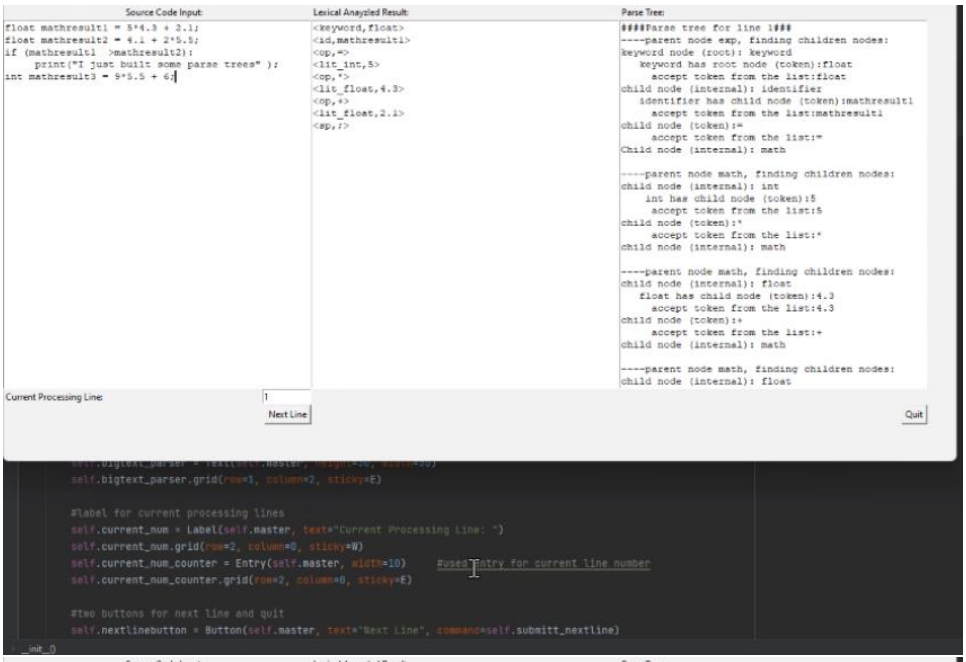
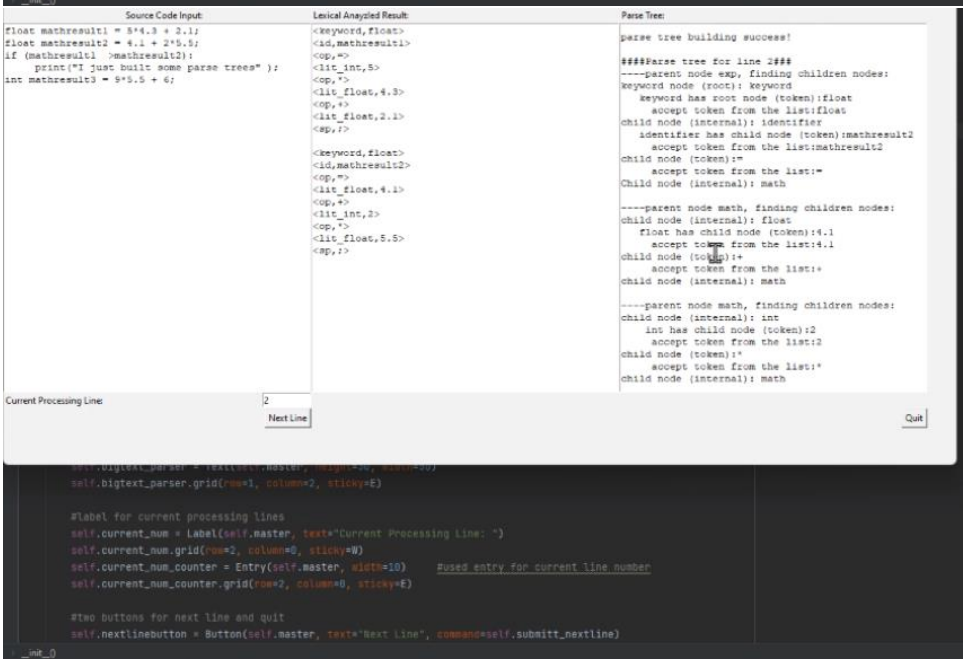
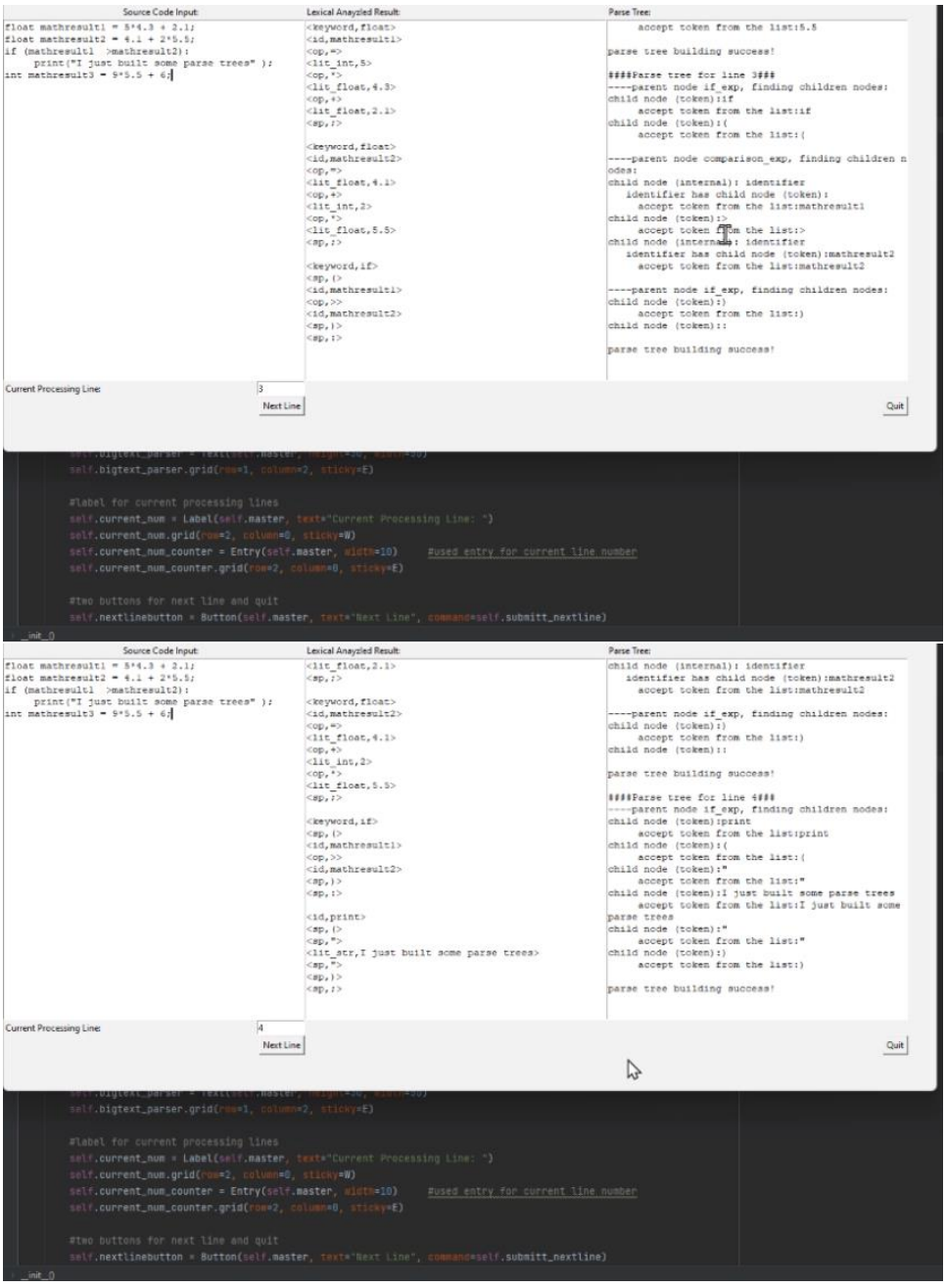
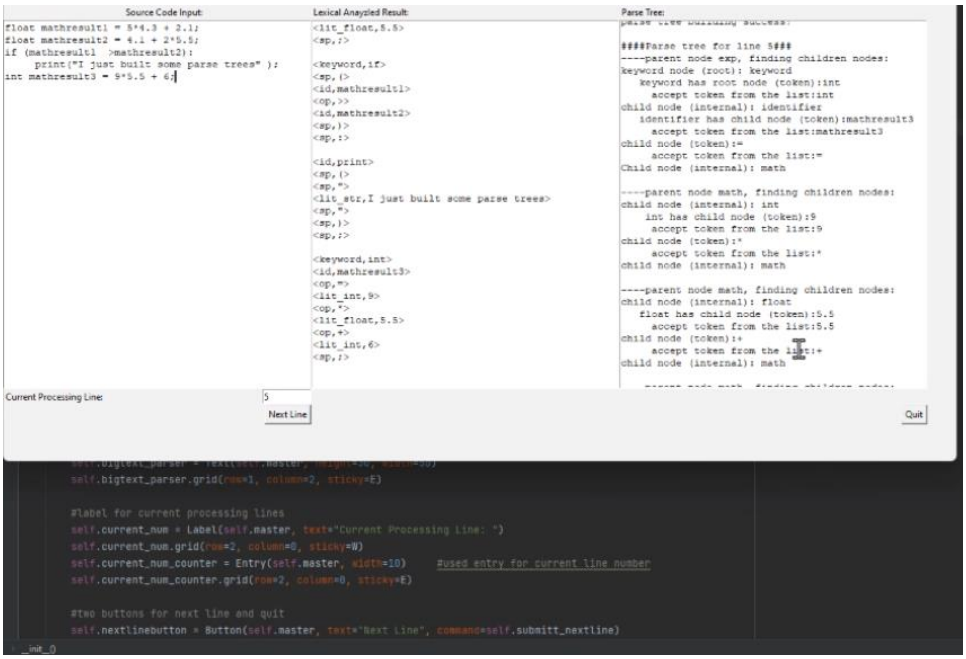

Source Code Input:
float mathresult1 = 5*4.3 + 2.1;
float mathresult2 = 4.1 + 2*5.5;
if (mathresult1 >mathresult2):
print("I just built some parse trees");
int mathresult3 = 9*5.5 + 6;
Lexical Anayzied Result:
<keyword,float>
<id,mathresult1>
<op,=>
<lit_int,5>
<op,*>
<lit_float,4.3>
<op,+>
<lit_float,2.1>
<sp,;>
Parse Tree:
####Parse tree for line 1###
----parent node exp, finding children nodes:
keyword node (root): keyword
keyword has root node (token):float
accept token from the list:float
child node (internal): identifier
identifier has child node (token):mathresult1
accept token from the list:mathresult1
child node (token):=
accept token from the list:=
Child node (internal): math
----parent node math, finding children nodes:
child node (internal): int
int has child node (token):5
accept token from the list:5
child node (token):*
accept token from the list:*
child node (internal): math
----parent node math, finding children nodes:
child node (internal): float
float has child node (token):4.3
accept token from the list:4.3
child node (token):+
accept token from the list:+
child node (internal): math
----parent node math, finding children nodes:
child node (internal): float
Current Processing Line:
1
Next Line
Quit
self.bigtext_parser.grid(row=1, column=2, sticky=E)
#label for current processing lines
self.current_num = Label(self.master, text="Current Processing Line: ")
self.current_num.grid(row=2, column=0, sticky=W)
self.current_num_counter = Entry(self.master, width=10)
#used entry for current line number
self.current_num_counter.grid(row=2, column=0, sticky=E)
#two buttons for next line and quit
self.nextlinebutton = Button(self.master, text="Next Line", command=self.submitt_nextline)
init()


Source Code Input:
float mathresult1 = 5*4.3 + 2.1;
float mathresult2 = 4.1 + 2*5.5;
if (mathresult1 >mathresult2):
print("I just built some parse trees");
int mathresult3 = 9*5.5 + 6;
Lexical Anayzied Result:
<keyword,float>
<id,mathresult1>
<op,=>
<lit_int,5>
<op,*>
<lit_float,4.3>
<op,+>
<lit_float,2.1>
<sp,;>
<keyword,float>
<id,mathresult2>
<op,=>
<lit_float,4.1>
<op,+>
<lit_int,2>
<op,*>
<lit_float,5.5>
<sp,;>
Parse Tree:
parse tree building success!
####Parse tree for line 2###
----parent node exp, finding children nodes:
keyword node (root): keyword
keyword has root node (token):float
accept token from the list:float
child node (internal): identifier
identifier has child node (token):mathresult2
accept token from the list:mathresult2
child node (token):=
accept token from the list:=
Child node (internal): math
----parent node math, finding children nodes:
child node (internal): float
float has child node (token):4.1
accept token from the list:4.1
child node (token):+
accept token from the list:+
child node (internal): math
----parent node math, finding children nodes:
child node (internal): int
int has child node (token):2
accept token from the list:2
child node (token):*
accept token from the list:*
child node (internal): math
Current Processing Line:
2
Next Line
Quit
self.bigtext_parser.grid(row=1, column=2, sticky=E)
#label for current processing lines
self.current_num = Label(self.master, text="Current Processing Line: ")
self.current_num.grid(row=2, column=0, sticky=W)
self.current_num_counter = Entry(self.master, width=10)
#used entry for current line number
self.current_num_counter.grid(row=2, column=0, sticky=E)
#two buttons for next line and quit
self.nextlinebutton = Button(self.master, text="Next Line", command=self.submitt_nextline)
init()


Source Code Input:
float mathresult1 = 5*4.3 + 2.1;
float mathresult2 = 4.1 + 2*5.5;
if (mathresult1 >mathresult2):
print("I just built some parse trees");
int mathresult3 = 9*5.5 + 6;
Lexical Anayzied Result:
<keyword,float>
<id,mathresult1>
<op,=>
<lit_int,5>
<op,*>
<lit_float,4.3>
<op,+>
<lit_float,2.1>
<sp,;>
<keyword,float>
<id,mathresult2>
<op,=>
<lit_float,4.1>
<op,+>
<lit_int,2>
<op,*>
<lit_float,5.5>
<sp,;>
<keyword,if>
<sp,(>
<id,mathresult1>
<op,>>
<id,mathresult2>
<sp,)>
<sp,:>
Parse Tree:
accept token from the list:5.5
parse tree building success!
####Parse tree for line 3###
----parent node if_exp, finding children nodes:
child node (token):if
accept token from the list:if
child node (token):(
accept token from the list:(
----parent node comparison_exp, finding children nodes:
child node (internal): identifier
identifier has child node (token):
accept token from the list:mathresult1
child node (token):>
accept token from the list:>
child node (internal): identifier
identifier has child node (token):mathresult2
accept token from the list:mathresult2
----parent node if_exp, finding children nodes:
child node (token):)
accept token from the list:)
child node (token)::
parse tree building success!
Current Processing Line:
3
Next Line
Quit
self.bigtext_parser.grid(row=1, column=2, sticky=E)
#label for current processing lines
self.current_num = Label(self.master, text="Current Processing Line: ")
self.current_num.grid(row=2, column=0, sticky=W)
self.current_num_counter = Entry(self.master, width=10) #used entry for current line number
self.current_num_counter.grid(row=2, column=0, sticky=E)
#two buttons for next line and quit
self.nextlinebutton = Button(self.master, text="Next Line", command=self.submitt_nextline)
init()

Source Code Input:
float mathresult1 = 5*4.3 + 2.1;
float mathresult2 = 4.1 + 2*5.5;
if (mathresult1 >mathresult2):
print("I just built some parse trees");
int mathresult3 = 9*5.5 + 6;
Lexical Anayzied Result:
<lit_float,2.1>
<sp,;>
<keyword,float>
<id,mathresult2>
<op,=>
<lit_float,4.1>
<op,+>
<lit_int,2>
<op,*>
<lit_float,5.5>
<sp,;>
<keyword,if>
<sp,(>
<id,mathresult1>
<op,>>
<id,mathresult2>
<sp,)>
<sp,:>
<id,print>
<sp,(>
<sp,">
<lit_str,I just built some parse trees>
<sp,">
<sp,)>
<sp,;>
Parse Tree:
child node (internal): identifier
identifier has child node (token):mathresult2
accept token from the list:mathresult2
----parent node if_exp, finding children nodes:
child node (token):)
accept token from the list:)
child node (token)::
parse tree building success!
####Parse tree for line 4###
----parent node if_exp, finding children nodes:
child node (token):print
accept token from the list:print
child node (token):(
accept token from the list:(
child node (token):"
accept token from the list:"
child node (token):I just built some parse trees
accept token from the list:I just built some parse trees
child node (token):"
accept token from the list:"
child node (token):)
accept token from the list:)
parse tree building success!
Current Processing Line:
4
Next Line
Quit
self.bigtext_parser.grid(row=1, column=2, sticky=E)
#label for current processing lines
self.current_num = Label(self.master, text="Current Processing Line: ")
self.current_num.grid(row=2, column=0, sticky=W)
self.current_num_counter = Entry(self.master, width=10) #used entry for current line number
self.current_num_counter.grid(row=2, column=0, sticky=E)
#two buttons for next line and quit
self.nextlinebutton = Button(self.master, text="Next Line", command=self.submitt_nextline)
init()


Source Code Input:
float mathresult1 = 5*4.3 + 2.1;
float mathresult2 = 4.1 + 2*5.5;
if (mathresult1 >mathresult2):
print("I just built some parse trees");
int mathresult3 = 5*5.5 + 6;
Lexical Anayzled Result:
<lit_float,5.5>
<sp,;>
<keyword,if>
<sp,(>
<id,mathresult1>
<op,>>
<id,mathresult2>
<sp,)>
<sp,:>
<id,print>
<sp,(>
<sp,">
<lit_str,I just built some parse trees>
<sp,">
<sp,)>
<sp,;>
<keyword,int>
<id,mathresult3>
<op,=>
<lit_int,9>
<op,*>
<lit_float,5.5>
<op,+>
<lit_int,6>
<sp,;>
Parse Tree:
####Parse tree for line 5###
----parent node exp, finding children nodes:
keyword node (root): keyword
keyword has root node (token)::int
accept token from the list:int
child node (internal): identifier
identifier has child node (token):mathresult3
accept token from the list:mathresult3
child node (token):=
accept token from the list:=
Child node (internal): math
----parent node math, finding children nodes:
child node (internal): int
int has child node (token):9
accept token from the list:9
child node (token):*
accept token from the list:*
child node (internal): math
----parent node math, finding children nodes:
child node (internal): float
float has child node (token):5.5
accept token from the list:5.5
child node (token):+
accept token from the list:+
child node (internal): math
Current Processing Line:
5
Next Line
Quit
self.bigtext_parser.grid(row=1, column=2, sticky=E)
#label for current processing lines
self.current_num = Label(self.master, text="Current Processing Line: ")
self.current_num.grid(row=2, column=0, sticky=W)
self.current_num_counter = Entry(self.master, width=10)
#used entry for current line number
self.current_num_counter.grid(row=2, column=0, sticky=E)
#two buttons for next line and quit
self.nextlinebutton = Button(self.master, text="Next Line", command=self.submitt_nextline)
__init__()