Lab 2 Parser

Sorin Macaluso

March 5, 2024

1 Test Cases

These are the test cases that I used for the program these are taken from Lexer without spaces and Gabriel Arnell Hall of Fame project. I have added changes to them to have unrecognized tokens and key words, as well a numbers in strings, pretty much anything that is not a lower case letter or space (since lowercase letters and space's are the only things allowed inside quotes). I have also included some smaller test cases to check for specific errors/events.

This will only show parser output and not the tokens since the tokens have been established already.

1.1 Test Case 1

```
1   /*valid 1*/
2   {
3   print(true)
4   print(false)
5   print(0)
6   print(7)
7   print("hello")
8   print(" ")
9   print("world")
10  print("")
11  print("end")
12  print(a)
13  }$
14
15  Paser starting :)
16  Parseing: Parse_Program()
17  Parseing: Parse_Block()
18  Parseing: Parse_Statement_List()
19  Parseing: Parse_Statement()
20  Parseing: Parse_Print_Statment()
21  Parseing: Parse_Expr()
22  Parseing: Parse_Boolean_Expr()
23  Parseing: Parse_Boolval()
24  Parseing: Parse_Statement_List()
25  Parseing: Parse_Statement()
26  Parseing: Parse_Print_Statment()
27  Parseing: Parse_Expr()
28  Parseing: Parse_Boolean_Expr()
29  Parseing: Parse_Boolval()
30  Parseing: Parse_Statement_List()
```

```
31  Parseing: Parse_Statement()
32  Parseing: Parse_Print_Statment()
33  Parseing: Parse_Expr()
34  Parseing: Parse_Int_Expr()
35  Parseing: Parse_Digit()
36  Parseing: Parse_Statement_List()
37  Parseing: Parse_Statement()
38  Parseing: Parse_Print_Statment()
39  Parseing: Parse_Expr()
40  Parseing: Parse_Int_Expr()
41  Parseing: Parse_Digit()
42  Parseing: Parse_Statement_List()
43  Parseing: Parse_Statement()
44  Parseing: Parse_Print_Statment()
45  Parseing: Parse_Expr()
46  Parseing: Parse_String_Expr()
47  Parseing: Parse_Char_List()
48  Parseing: Parse_Char()
49  Parseing: Parse_Char_List()
50  Parseing: Parse_Char()
51  Parseing: Parse_Char_List()
52  Parseing: Parse_Char()
53  Parseing: Parse_Char_List()
54  Parseing: Parse_Char()
55  Parseing: Parse_Char_List()
56  Parseing: Parse_Char()
57  Parseing: Parse_Char_List()
58  Parseing: Parse_Statement_List()
59  Parseing: Parse_Statement()
60  Parseing: Parse_Print_Statment()
61  Parseing: Parse_Expr()
62  Parseing: Parse_String_Expr()
63  Parseing: Parse_Char_List()
64  Parseing: Parse_Space()
65  Parseing: Parse_Char_List()
66  Parseing: Parse_Statement_List()
67  Parseing: Parse_Statement()
68  Parseing: Parse_Print_Statment()
69  Parseing: Parse_Expr()
70  Parseing: Parse_String_Expr()
71  Parseing: Parse_Char_List()
72  Parseing: Parse_Char()
73  Parseing: Parse_Char_List()
74  Parseing: Parse_Char()
75  Parseing: Parse_Char_List()
76  Parseing: Parse_Char()
77  Parseing: Parse_Char_List()
78  Parseing: Parse_Char()
79  Parseing: Parse_Char_List()
80  Parseing: Parse_Char()
81  Parseing: Parse_Char_List()
82  Parseing: Parse_Statement_List()
83  Parseing: Parse_Statement()
84  Parseing: Parse_Print_Statment()
```

```
 85  Parseing: Parse_Expr()
 86  Parseing: Parse_String_Expr()
 87  Parseing: Parse_Char_List()
 88  Parseing: Parse_Statement_List()
 89  Parseing: Parse_Statement()
 90  Parseing: Parse_Print_Statment()
 91  Parseing: Parse_Expr()
 92  Parseing: Parse_String_Expr()
 93  Parseing: Parse_Char_List()
 94  Parseing: Parse_Char()
 95  Parseing: Parse_Char_List()
 96  Parseing: Parse_Char()
 97  Parseing: Parse_Char_List()
 98  Parseing: Parse_Char()
 99  Parseing: Parse_Char_List()
100  Parseing: Parse_Statement_List()
101  Parseing: Parse_Statement()
102  Parseing: Parse_Print_Statment()
103  Parseing: Parse_Expr()
104  Parseing: Parse_Id()
105  Parseing: Parse_Char()
106  Parseing: Parse_Statement_List()
107  Parse for Program 1 Done
108  Parse No errors :)
109  CST will be displayed :)
110  <program>
111
112  -<block>
113
114  --[{]
115
116  --<statement_list>
117
118  ---<statement>
119
120  ----<print_statment>
121
122  -----[print]
123
124  -----[(]
125
126  -----<expr>
127
128  ------<boolean_expr>
129
130  -------<boolval>
131
132  --------[true]
133
134  -----[)]
135
136  ---<statement_list>
137
138  ----<statement>
```

```
139
140    -----<print_statment>
141
142    ------[print]
143
144    ------[(]
145
146    ------<expr>
147
148    -------<boolean_expr>
149
150    --------<boolval>
151
152    ---------[false]
153
154    ------[)]
155
156    ----<statement_list>
157
158    -----<statement>
159
160    ------<print_statment>
161
162    -------[print]
163
164    -------[(]
165
166    -------<expr>
167
168    --------<int_expr>
169
170    ---------<digit>
171
172    ----------[0]
173
174    -------[)]
175
176    -----<statement_list>
177
178    ------<statement>
179
180    -------<print_statment>
181
182    -------[print]
183
184    --------[(]
185
186    --------<expr>
187
188    ---------<int_expr>
189
190    ----------<digit>
191
192    -----------[7]
```

```
193
194  --------[)]
195
196  ------<statement_list>
197
198  -------<statement>
199
200  --------<print_statment>
201
202  ---------[print]
203
204  ---------[(]
205
206  ---------<expr>
207
208  ----------<string_expr>
209
210  -----------["]
211
212  -----------<char_list>
213
214  -----------<char>
215
216  ------------[h]
217
218  ------------<char_list>
219
220  ------------<char>
221
222  -------------[e]
223
224  -------------<char_list>
225
226  -------------<char>
227
228  --------------[l]
229

230  --------------<char_list>
231
232  --------------<char>
233
234  ---------------[l]
235
236  ---------------<char_list>
237
238  ---------------<char>
239
240  ----------------[o]
241
242  ----------------[char_list]
243
244  -----------["]
245
246  ---------[)]
```

```
--------<statement_list>

--------<statement>

---------<print_statment>

----------[print]

----------[(]

----------<expr>

-----------<string_expr>

------------["]

------------<char_list>

-------------<space>

--------------[ ]

-------------[char_list]

------------["]

----------[)]

--------<statement_list>

---------<statement>

----------<print_statment>

-----------[print]

-----------[(]

-----------<expr>

------------<string_expr>

-------------["]

-------------<char_list>

--------------<char>

---------------[w]

--------------<char_list>

---------------<char>
```

```
---------------[o]

--------------<char_list>

--------------<char>

----------------[r]

---------------<char_list>

---------------<char>

-----------------[l]

----------------<char_list>

-----------------<char>

------------------[d]

-----------------[char_list]

-------------["]

-----------[)]

---------<statement_list>

----------<statement>

-----------<print_statment>

-----------[print]

------------[(]

------------<expr>

-------------<string_expr>

--------------["]

--------------[char_list]

--------------["]

------------[)]

----------<statement_list>

-----------<statement>

------------<print_statment>
```

```
355
356   -------------[print]
357
358   -------------[(]
359
360   -------------<expr>
361
362   --------------<string_expr>
363
364   --------------["]
365
366   --------------<char_list>
367
368   ---------------<char>
369
370   ----------------[e]
371
372   ---------------<char_list>
373
374   ----------------<char>
375
376   -----------------[n]
377
378   ----------------<char_list>
379
380   -----------------<char>
381
382   ------------------[d]
383
384   -----------------[char_list]
385
386   --------------["]
387
388   -------------[)]
389
390   -----------<statement_list>
391
392   ------------<statement>
393
394   ------------<print_statment>
395
396   -------------[print]
397
398   -------------[(]
399
400   -------------<expr>
401
402   -------------<id>
403
404   ---------------<char>
405
406   ----------------[a]
407
408   -------------[)]
```

```
409
410  ------------[statement_list]
411
412  --[}]
413
414  -[$]
415
416  End of program 1
```

1.2 Test Case 2

```
1  /*valid 2*/
2  {}$
3
4  Paser starting :)
5  Parseing: Parse_Program()
6  Parseing: Parse_Block()
7  Parseing: Parse_Statement_List()
8  Parse for Program 2 Done
9  Parse No errors :)
10  CST will be displayed :)
11  <program>
12
13  -<block>
14
15  --[{]
16
17  --[statement_list]
18
19  --[}]
20
21  -[$]
22
23  End of program 2
```

1.3 Test Case 3

```
1  /*valid 3*/
2  {{{}}}$
3
4  Paser starting :)
5  Parseing: Parse_Program()
6  Parseing: Parse_Block()
7  Parseing: Parse_Statement_List()
8  Parseing: Parse_Statement()
9  Parseing: Parse_Block()
10  Parseing: Parse_Statement_List()
11  Parseing: Parse_Statement()
12  Parseing: Parse_Block()
13  Parseing: Parse_Statement_List()
14  Parseing: Parse_Statement_List()
15  Parseing: Parse_Statement_List()
```

```
16  Parse for Program 3 Done
17  Parse No errors :)
18  CST will be displayed :)
19  <program>
20
21  -<block>
22
23  --[{]
24
25  --<statement_list>
26
27  ---<statement>
28
29  ----<block>
30
31  -----[{]
32
33  -----<statement_list>
34
35  ------<statement>
36
37  -------<block>
38
39  --------[{]
40
41  --------[statement_list]
42
43  --------[}]
44
45  ------[statement_list]
46
47  -----[}]
48
49  ---[statement_list]
50
51  --[}]
52
53  -[$]
54
55  End of program 3
```

1.4   Test Case 4

```
1   /*valid 4*/
2   /*Long Test Case - Everything Except Boolean Declaration */
3   {
4          /* Int Declaration */
5          int a
6          int b
7          a = 0
8          b=0
9          /* While Loop */
10         while (a != 3) {
```

```
11              print(a)
12              while (b != 3) {
13                  print(b)
14                  b = 1 + b
15                  if (b == 2) {
16                      /* Print Statement */
17                      print("there is no spoon" /* This will do nothing */ )
18                  }
19              }
20              b = 0
21              a = 1+a
22          }
23      }
24  $
25
26  Paser starting :)
27  Parseing: Parse_Program()
28  Parseing: Parse_Block()
29  Parseing: Parse_Statement_List()
30  Parseing: Parse_Statement()
31  Parseing: Parse_Var_Decl()
32  Parseing: Parse_Type()
33  Parseing: Parse_Id()
34  Parseing: Parse_Char()
35  Parseing: Parse_Statement_List()
36  Parseing: Parse_Statement()
37  Parseing: Parse_Var_Decl()
38  Parseing: Parse_Type()
39  Parseing: Parse_Id()
40  Parseing: Parse_Char()
41  Parseing: Parse_Statement_List()
42  Parseing: Parse_Statement()
43  Parseing: Parse_Assignment_Statment()
44  Parseing: Parse_Id()
45  Parseing: Parse_Char()
46  Parseing: Parse_Expr()
47  Parseing: Parse_Int_Expr()
48  Parseing: Parse_Digit()
49  Parseing: Parse_Statement_List()
50  Parseing: Parse_Statement()
51  Parseing: Parse_Assignment_Statment()
52  Parseing: Parse_Id()
53  Parseing: Parse_Char()
54  Parseing: Parse_Expr()
55  Parseing: Parse_Int_Expr()
56  Parseing: Parse_Digit()
57  Parseing: Parse_Statement_List()
58  Parseing: Parse_Statement()
59  Parseing: Parse_While_Statment()
60  Parseing: Parse_Boolean_Expr()
61  Parseing: Parse_Expr()
62  Parseing: Parse_Id()
63  Parseing: Parse_Char()
64  Parseing: Parse_Boolop()
```

```
65    Parseing:  Parse_Expr ()
66    Parseing:  Parse_Int_Expr ()
67    Parseing:  Parse_Digit ()
68    Parseing:  Parse_Block ()
69    Parseing:  Parse_Statement_List ()
70    Parseing:  Parse_Statement ()
71    Parseing:  Parse_Print_Statment ()
72    Parseing:  Parse_Expr ()
73    Parseing:  Parse_Id ()
74    Parseing:  Parse_Char ()
75    Parseing:  Parse_Statement_List ()
76    Parseing:  Parse_Statement ()
77    Parseing:  Parse_While_Statment ()
78    Parseing:  Parse_Boolean_Expr ()
79    Parseing:  Parse_Expr ()
80    Parseing:  Parse_Id ()
81    Parseing:  Parse_Char ()
82    Parseing:  Parse_Boolop ()
83    Parseing:  Parse_Expr ()
84    Parseing:  Parse_Int_Expr ()
85    Parseing:  Parse_Digit ()
86    Parseing:  Parse_Block ()
87    Parseing:  Parse_Statement_List ()
88    Parseing:  Parse_Statement ()
89    Parseing:  Parse_Print_Statment ()
90    Parseing:  Parse_Expr ()
91    Parseing:  Parse_Id ()
92    Parseing:  Parse_Char ()
93    Parseing:  Parse_Statement_List ()
94    Parseing:  Parse_Statement ()
95    Parseing:  Parse_Assignment_Statment ()
96    Parseing:  Parse_Id ()
97    Parseing:  Parse_Char ()
98    Parseing:  Parse_Expr ()
99    Parseing:  Parse_Int_Expr ()
100   Parseing:  Parse_Digit ()
101   Parseing:  Parse_intop ()
102   Parseing:  Parse_Expr ()
103   Parseing:  Parse_Id ()
104   Parseing:  Parse_Char ()
105   Parseing:  Parse_Statement_List ()
106   Parseing:  Parse_Statement ()
107   Parseing:  Parse_If_Statment ()
108   Parseing:  Parse_Boolean_Expr ()
109   Parseing:  Parse_Expr ()
110   Parseing:  Parse_Id ()
111   Parseing:  Parse_Char ()
112   Parseing:  Parse_Boolop ()
113   Parseing:  Parse_Expr ()
114   Parseing:  Parse_Int_Expr ()
115   Parseing:  Parse_Digit ()
116   Parseing:  Parse_Block ()
117   Parseing:  Parse_Statement_List ()
118   Parseing:  Parse_Statement ()
```

```
119  Parseing:  Parse_Print_Statment ()
120  Parseing:  Parse_Expr ()
121  Parseing:  Parse_String_Expr ()
122  Parseing:  Parse_Char_List ()
123  Parseing:  Parse_Char ()
124  Parseing:  Parse_Char_List ()
125  Parseing:  Parse_Char ()
126  Parseing:  Parse_Char_List ()
127  Parseing:  Parse_Char ()
128  Parseing:  Parse_Char_List ()
129  Parseing:  Parse_Char ()
130  Parseing:  Parse_Char_List ()
131  Parseing:  Parse_Char ()
132  Parseing:  Parse_Char_List ()
133  Parseing:  Parse_Space ()
134  Parseing:  Parse_Char_List ()
135  Parseing:  Parse_Char ()
136  Parseing:  Parse_Char_List ()
137  Parseing:  Parse_Char ()
138  Parseing:  Parse_Char_List ()
139  Parseing:  Parse_Space ()
140  Parseing:  Parse_Char_List ()
141  Parseing:  Parse_Char ()
142  Parseing:  Parse_Char_List ()
143  Parseing:  Parse_Char ()
144  Parseing:  Parse_Char_List ()
145  Parseing:  Parse_Space ()
146  Parseing:  Parse_Char_List ()
147  Parseing:  Parse_Char ()
148  Parseing:  Parse_Char_List ()
149  Parseing:  Parse_Char ()
150  Parseing:  Parse_Char_List ()
151  Parseing:  Parse_Char ()
152  Parseing:  Parse_Char_List ()
153  Parseing:  Parse_Char ()
154  Parseing:  Parse_Char_List ()
155  Parseing:  Parse_Char ()
156  Parseing:  Parse_Char_List ()
157  Parseing:  Parse_Statement_List ()
158  Parseing:  Parse_Statement_List ()
159  Parseing:  Parse_Statement_List ()
160  Parseing:  Parse_Statement ()
161  Parseing:  Parse_Assignment_Statment ()
162  Parseing:  Parse_Id ()
163  Parseing:  Parse_Char ()
164  Parseing:  Parse_Expr ()
165  Parseing:  Parse_Int_Expr ()
166  Parseing:  Parse_Digit ()
167  Parseing:  Parse_Statement_List ()
168  Parseing:  Parse_Statement ()
169  Parseing:  Parse_Assignment_Statment ()
170  Parseing:  Parse_Id ()
171  Parseing:  Parse_Char ()
172  Parseing:  Parse_Expr ()
```

```
173  Parseing: Parse_Int_Expr()
174  Parseing: Parse_Digit()
175  Parseing: Parse_intop()
176  Parseing: Parse_Expr()
177  Parseing: Parse_Id()
178  Parseing: Parse_Char()
179  Parseing: Parse_Statement_List()
180  Parseing: Parse_Statement_List()
181  Parse for Program 4 Done
182  Parse No errors :)
183  CST will be displayed :)
184  <program>
185
186  -<block>
187
188  --[{]
189
190  --<statement_list>
191
192  ---<statement>
193
194  ----<var_decl>
195
196  -----<type>
197
198  ------[int]
199
200  -----<id>
201
202  ------<char>
203
204  -------[a]
205
206  ---<statement_list>
207
208  ----<statement>
209
210  -----<var_decl>
211
212  ------<type>
213
214  -------[int]
215
216  ------<id>
217
218  -------<char>
219

220  --------[b]
221

222  ----<statement_list>
223

224  -----<statement>
225

226  ------<assignment_statment>
```

```
227
228  -------<id>
229
230  -------<char>
231
232  ---------[a]
233
234  -------[=]
235
236  -------<expr>
237
238  -------<int_expr>
239
240  ---------<digit>
241
242  ----------[0]
243
244  -----<statement_list>
245
246  ------<statement>
247
248  -------<assignment_statment>
249
250  --------<id>
251
252  ---------<char>
253
254  ----------[b]
255
256  -------[=]
257
258  --------<expr>
259
260  ---------<int_expr>
261
262  ----------<digit>
263
264  -----------[0]
265
266  ------<statement_list>
267
268  -------<statement>
269
270  -------<while_statment>
271
272  ---------[while]
273
274  --------<boolean_expr>
275
276  ----------[(]
277
278  ----------<expr>
279
280  -----------<id>
```

```
281
282   ------------<char>
283
284   ------------[a]
285
286   ----------<boolop>
287
288   ----------[!=]
289
290   ----------<expr>
291
292   ----------<int_expr>
293
294   ------------<digit>
295
296   ------------[3]
297
298   ----------[)]
299
300   ---------<block>
301
302   ----------[{]
303
304   ----------<statement_list>
305
306   ----------<statement>
307
308   -----------<print_statment>
309
310   ------------[print]
311
312   ------------[(]
313
314   ------------<expr>
315
316   -------------<id>
317
318   --------------<char>
319
320   ---------------[a]
321
322   ------------[)]
323
324   ----------<statement_list>
325
326   -----------<statement>
327
328   ------------<while_statment>
329
330   -------------[while]
331
332   -------------<boolean_expr>
333
334   --------------[(]
```

```
335
336  ---------------<expr>
337
338  ---------------<id>
339
340  ----------------<char>
341
342  -----------------[b]
343
344  --------------<boolop>
345
346  ---------------[!=]
347
348  --------------<expr>
349
350  ---------------<int_expr>
351
352  ----------------<digit>
353
354  -----------------[3]
355
356  --------------[)]
357
358  -------------<block>
359
360  --------------[{]
361
362  --------------<statement_list>
363
364  ---------------<statement>
365
366  ----------------<print_statment>
367
368  -----------------[print]
369
370  -----------------[(]
371
372  -----------------<expr>
373
374  ------------------<id>
375
376  -------------------<char>
377
378  --------------------[b]
379
380  -----------------[)]
381
382  --------------<statement_list>
383
384  ---------------<statement>
385
386  ----------------<assignment_statment>
387
388  -----------------<id>
```

```
389
390   ------------------<char>
391
392   ------------------[b]
393
394   ------------------[=]
395
396   ------------------<expr>
397
398   ------------------<int_expr>
399
400   ------------------<digit>
401
402   ------------------[1]
403
404   ------------------<intop>
405
406   ------------------[+]
407
408   ------------------<expr>
409
410   ------------------<id>
411
412   --------------------<char>
413
414   --------------------[b]
415
416   ----------------<statement_list>
417
418   ----------------<statement>
419
420   ------------------<if_statment>
421
422   ------------------[if]
423
424   ------------------<boolean_expr>
425
426   --------------------[(]
427
428   --------------------<expr>
429
430   --------------------<id>
431
432   --------------------<char>
433
434   ----------------------[b]
435
436   --------------------<boolop>
437
438   --------------------[==]
439
440   --------------------<expr>
441
442   --------------------<int_expr>
```

```
443
444  ----------------------<digit>
445
446  ----------------------[2]
447
448  --------------------[)]
449
450  ------------------<block>
451
452  --------------------[{]
453
454  -------------------<statement_list>
455
456  --------------------<statement>
457
458  ---------------------<print_statment>
459
460  ----------------------[print]
461
462  ----------------------[(]
463
464  ----------------------<expr>
465
466  -----------------------<string_expr>
467
468  ------------------------["]
469
470  ------------------------<char_list>
471
472  -------------------------<char>
473
474  --------------------------[t]
475
476  -------------------------<char_list>
477
478  --------------------------<char>
479

480  ---------------------------[h]
481
482  --------------------------<char_list>
483
484  ---------------------------<char>
485
486  ----------------------------[e]
487
488  ---------------------------<char_list>
489
490  ----------------------------<char>
491
492  -----------------------------[r]
493
494  ----------------------------<char_list>
495
496  -----------------------------<char>
```

```
497
498  ------------------------------[e]
499
500  -----------------------------<char_list>
501
502  -----------------------------<space>
503
504  -----------------------------[ ]
505
506  -----------------------------<char_list>
507
508  -----------------------------<char>
509
510  ------------------------------[i]
511
512  ------------------------------<char_list>
513
514  ------------------------------<char>
515
516  ------------------------------[s]
517
518  -------------------------------<char_list>
519
520  -------------------------------<space>
521
522  -------------------------------[ ]
523
524  --------------------------------<char_list>
525
526  --------------------------------<char>
527
528  --------------------------------[n]
529
530  ---------------------------------<char_list>
531
532  ---------------------------------<char>
533
534  ---------------------------------[o]
535
536  ----------------------------------<char_list>
537
538  ----------------------------------<space>
539
540  ----------------------------------[ ]
541
542  -----------------------------------<char_list>
543
544  -----------------------------------<char>
545
546  -----------------------------------[s]
547
548  ------------------------------------<char_list>
549
550  ------------------------------------<char>
```

```
551
552  ---------------------------------------[p]
553
554  ---------------------------------------<char_list>
555
556  ---------------------------------------<char>
557
558  ---------------------------------------[o]
559
560  ---------------------------------------<char_list>
561
562  ---------------------------------------<char>
563
564  ---------------------------------------[o]
565
566  ---------------------------------------<char_list>
567
568  ---------------------------------------<char>
569
570  ---------------------------------------[n]
571
572  ---------------------------------------[char_list]
573
574  -------------------------["]
575
576  ----------------------[)]
577
578  --------------------[statement_list]
579
580  -------------------[}]
581
582  -----------------[statement_list]
583
584  --------------[}]
585
586  -----------<statement_list>
587
588  -------------<statement>
589
590  -------------<assignment_statment>
591
592  --------------<id>
593
594  ---------------<char>
595
596  ----------------[b]
597
598  --------------[=]
599
600  --------------<expr>
601
602  ----------------<int_expr>
603
604  -----------------<digit>
```

```
605
606  -----------------[0]
607
608  -------------<statement_list>
609
610  -------------<statement>
611
612  --------------<assignment_statment>
613
614  ---------------<id>
615
616  ----------------<char>
617
618  -----------------[a]
619
620  ---------------[=]
621
622  ---------------<expr>
623
624  ----------------<int_expr>
625
626  -----------------<digit>
627
628  ------------------[1]
629
630  -----------------<intop>
631
632  ------------------[+]
633
634  -----------------<expr>
635
636  ------------------<id>
637
638  -------------------<char>
639
640  --------------------[a]
641
642  -------------[statement_list]
643
644  ----------[}]
645
646  -------[statement_list]
647
648  --[}]
649
650  -[$]
651
652  End of program 4
```

1.5 Test Case 5

```
1  /*valid 5*/
2  {
```

```
 3              int a
 4              boolean b
 5              string c
 6          a = 9
 7           b = true
 8           {
 9               print(a)
10               print(b)
11               b = false
12               c = "hello world"
13               int b
14               b = 0
15               {
16                   print(c)
17                   a = 1 + 2
18                   {
19                       print(b)
20                   }
21               }
22               b = a
23               print(b)
24           }
25           print(b)
26  }$
27
28  Paser starting :)
29  Parseing: Parse_Program()
30  Parseing: Parse_Block()
31  Parseing: Parse_Statement_List()
32  Parseing: Parse_Statement()
33  Parseing: Parse_Var_Decl()
34  Parseing: Parse_Type()
35  Parseing: Parse_Id()
36  Parseing: Parse_Char()
37  Parseing: Parse_Statement_List()
38  Parseing: Parse_Statement()
39  Parseing: Parse_Var_Decl()
40  Parseing: Parse_Type()
41  Parseing: Parse_Id()
42  Parseing: Parse_Char()
43  Parseing: Parse_Statement_List()
44  Parseing: Parse_Statement()
45  Parseing: Parse_Var_Decl()
46  Parseing: Parse_Type()
47  Parseing: Parse_Id()
48  Parseing: Parse_Char()
49  Parseing: Parse_Statement_List()
50  Parseing: Parse_Statement()
51  Parseing: Parse_Assignment_Statment()
52  Parseing: Parse_Id()
53  Parseing: Parse_Char()
54  Parseing: Parse_Expr()
55  Parseing: Parse_Int_Expr()
56  Parseing: Parse_Digit()
```

```
57  Parseing: Parse_Statement_List ()
58  Parseing: Parse_Statement ()
59  Parseing: Parse_Assignment_Statment ()
60  Parseing: Parse_Id ()
61  Parseing: Parse_Char ()
62  Parseing: Parse_Expr ()
63  Parseing: Parse_Boolean_Expr ()
64  Parseing: Parse_Boolval ()
65  Parseing: Parse_Statement_List ()
66  Parseing: Parse_Statement ()
67  Parseing: Parse_Block ()
68  Parseing: Parse_Statement_List ()
69  Parseing: Parse_Statement ()
70  Parseing: Parse_Print_Statment ()
71  Parseing: Parse_Expr ()
72  Parseing: Parse_Id ()
73  Parseing: Parse_Char ()
74  Parseing: Parse_Statement_List ()
75  Parseing: Parse_Statement ()
76  Parseing: Parse_Print_Statment ()
77  Parseing: Parse_Expr ()
78  Parseing: Parse_Id ()
79  Parseing: Parse_Char ()
80  Parseing: Parse_Statement_List ()
81  Parseing: Parse_Statement ()
82  Parseing: Parse_Assignment_Statment ()
83  Parseing: Parse_Id ()
84  Parseing: Parse_Char ()
85  Parseing: Parse_Expr ()
86  Parseing: Parse_Boolean_Expr ()
87  Parseing: Parse_Boolval ()
88  Parseing: Parse_Statement_List ()
89  Parseing: Parse_Statement ()
90  Parseing: Parse_Assignment_Statment ()
91  Parseing: Parse_Id ()
92  Parseing: Parse_Char ()
93  Parseing: Parse_Expr ()
94  Parseing: Parse_String_Expr ()
95  Parseing: Parse_Char_List ()
96  Parseing: Parse_Char ()
97  Parseing: Parse_Char_List ()
98  Parseing: Parse_Char ()
99  Parseing: Parse_Char_List ()
100 Parseing: Parse_Char ()
101 Parseing: Parse_Char_List ()
102 Parseing: Parse_Char ()
103 Parseing: Parse_Char_List ()
104 Parseing: Parse_Char ()
105 Parseing: Parse_Char_List ()
106 Parseing: Parse_Space ()
107 Parseing: Parse_Char_List ()
108 Parseing: Parse_Char ()
109 Parseing: Parse_Char_List ()
110 Parseing: Parse_Char ()
```

```
111  Parseing: Parse_Char_List()
112  Parseing: Parse_Char()
113  Parseing: Parse_Char_List()
114  Parseing: Parse_Char()
115  Parseing: Parse_Char_List()
116  Parseing: Parse_Char()
117  Parseing: Parse_Char_List()
118  Parseing: Parse_Statement_List()
119  Parseing: Parse_Statement()
120  Parseing: Parse_Var_Decl()
121  Parseing: Parse_Type()
122  Parseing: Parse_Id()
123  Parseing: Parse_Char()
124  Parseing: Parse_Statement_List()
125  Parseing: Parse_Statement()
126  Parseing: Parse_Assignment_Statment()
127  Parseing: Parse_Id()
128  Parseing: Parse_Char()
129  Parseing: Parse_Expr()
130  Parseing: Parse_Int_Expr()
131  Parseing: Parse_Digit()
132  Parseing: Parse_Statement_List()
133  Parseing: Parse_Statement()
134  Parseing: Parse_Block()
135  Parseing: Parse_Statement_List()
136  Parseing: Parse_Statement()
137  Parseing: Parse_Print_Statment()
138  Parseing: Parse_Expr()
139  Parseing: Parse_Id()
140  Parseing: Parse_Char()
141  Parseing: Parse_Statement_List()
142  Parseing: Parse_Statement()
143  Parseing: Parse_Assignment_Statment()
144  Parseing: Parse_Id()
145  Parseing: Parse_Char()
146  Parseing: Parse_Expr()
147  Parseing: Parse_Int_Expr()
148  Parseing: Parse_Digit()
149  Parseing: Parse_intop()
150  Parseing: Parse_Expr()
151  Parseing: Parse_Int_Expr()
152  Parseing: Parse_Digit()
153  Parseing: Parse_Statement_List()
154  Parseing: Parse_Statement()
155  Parseing: Parse_Block()
156  Parseing: Parse_Statement_List()
157  Parseing: Parse_Statement()
158  Parseing: Parse_Print_Statment()
159  Parseing: Parse_Expr()
160  Parseing: Parse_Id()
161  Parseing: Parse_Char()
162  Parseing: Parse_Statement_List()
163  Parseing: Parse_Statement_List()
164  Parseing: Parse_Statement_List()
```

```
165  Parseing: Parse_Statement()
166  Parseing: Parse_Assignment_Statment()
167  Parseing: Parse_Id()
168  Parseing: Parse_Char()
169  Parseing: Parse_Expr()
170  Parseing: Parse_Id()
171  Parseing: Parse_Char()
172  Parseing: Parse_Statement_List()
173  Parseing: Parse_Statement()
174  Parseing: Parse_Print_Statment()
175  Parseing: Parse_Expr()
176  Parseing: Parse_Id()
177  Parseing: Parse_Char()
178  Parseing: Parse_Statement_List()
179  Parseing: Parse_Statement_List()
180  Parseing: Parse_Statement()
181  Parseing: Parse_Print_Statment()
182  Parseing: Parse_Expr()
183  Parseing: Parse_Id()
184  Parseing: Parse_Char()
185  Parseing: Parse_Statement_List()
186  Parse for Program 5 Done
187  Parse No errors :)
188  CST will be displayed :)
189  <program>
190
191  -<block>
192
193  --[{]
194
195  --<statement_list>
196
197  ---<statement>
198
199  ----<var_decl>
200
201  -----<type>
202
203  ------[int]
204
205  -----<id>
206
207  ------<char>
208
209  -------[a]
210
211  ---<statement_list>
212
213  ----<statement>
214
215  -----<var_decl>
216
217  ------<type>
218
```

```
-------[boolean]

------<id>

-------<char>

--------[b]

----<statement_list>

-----<statement>

------<var_decl>

-------<type>

--------[string]

-------<id>

--------<char>

---------[c]

-----<statement_list>

------<statement>

-------<assignment_statment>

--------<id>

---------<char>

----------[a]

-------[=]

--------<expr>

--------<int_expr>

---------<digit>

----------[9]

------<statement_list>

-------<statement>

-------<assignment_statment>

---------<id>
```

```
273  ----------<char>

274

275  -----------[b]

276

277  ---------[=]

278

279  ---------<expr>

280

281  ----------<boolean_expr>

282

283  -----------<boolval>

284

285  ------------[true]

286

287  -------<statement_list>

288

289  --------<statement>

290

291  ---------<block>

292

293  ----------[{]

294

295  ----------<statement_list>

296

297  -----------<statement>

298

299  -----------<print_statment>

300

301  ------------[print]

302

303  ------------[(]

304

305  ------------<expr>

306

307  -------------<id>

308

309  --------------<char>

310

311  ---------------[a]

312

313  ------------[)]

314

315  ----------<statement_list>

316

317  -----------<statement>

318

319  ------------<print_statment>

320

321  -------------[print]

322

323  -------------[(]

324

325  --------------<expr>

326
```

```
327   ---------------<id>

328

329   ---------------<char>

330

331   ---------------[b]

332

333   -------------[)]

334

335   -----------<statement_list>

336

337   ------------<statement>

338

339   -------------<assignment_statment>

340

341   --------------<id>

342

343   ---------------<char>

344

345   ---------------[b]

346

347   --------------[=]

348

349   --------------<expr>

350

351   ---------------<boolean_expr>

352

353   ---------------<boolval>

354

355   ----------------[false]

356

357   ------------<statement_list>

358

359   -------------<statement>

360

361   --------------<assignment_statment>

362

363   ---------------<id>

364

365   ----------------<char>

366

367   ----------------[c]

368

369   ---------------[=]

370

371   --------------<expr>

372

373   ---------------<string_expr>

374

375   ----------------["]

376

377   ----------------<char_list>

378

379   -----------------<char>

380
```

```
381  -------------------[h]
382
383  -------------------<char_list>
384
385  -------------------<char>
386
387  -------------------[e]
388
389  -------------------<char_list>
390
391  -------------------<char>
392
393  -------------------[l]
394
395  -------------------<char_list>
396
397  --------------------<char>
398
399  --------------------[l]
400
401  --------------------<char_list>
402
403  --------------------<char>
404
405  ---------------------[o]
406
407  ---------------------<char_list>
408
409  ---------------------<space>
410
411  ----------------------[ ]
412
413  ----------------------<char_list>
414
415  -----------------------<char>
416
417  -----------------------[w]
418
419  -----------------------<char_list>
420
421  ------------------------<char>
422
423  ------------------------[o]
424
425  ------------------------<char_list>
426
427  -------------------------<char>
428
429  -------------------------[r]
430
431  -------------------------<char_list>
432
433  --------------------------<char>
434
```

```
435   ---------------------------[l]

436

437   --------------------------<char_list>

438

439   --------------------------<char>

440

441   ---------------------------[d]

442

443   --------------------------[char_list]

444

445   -----------------["]

446

447   --------------<statement_list>

448

449   --------------<statement>

450

451   ---------------<var_decl>

452

453   ---------------<type>

454

455   -----------------[int]

456

457   ---------------<id>

458

459   -----------------<char>

460

461   -----------------[b]

462

463   --------------<statement_list>

464

465   ---------------<statement>

466

467   ----------------<assignment_statment>

468

469   ----------------<id>

470

471   -----------------<char>

472

473   ------------------[b]

474

475   -----------------[=]

476

477   -----------------<expr>

478

479   ------------------<int_expr>

480

481   ------------------<digit>

482

483   -------------------[0]

484

485   --------------<statement_list>

486

487   ---------------<statement>

488
```

```
489 ------------------<block>
490
491 -------------------[{]
492
493 ------------------<statement_list>
494
495 -------------------<statement>
496
497 --------------------<print_statment>
498
499 ---------------------[print]
500
501 ---------------------[(]
502
503 ---------------------<expr>
504
505 ----------------------<id>
506
507 -----------------------<char>
508
509 ------------------------[c]
510
511 ---------------------[)]
512
513 --------------------<statement_list>
514
515 --------------------<statement>
516
517 ---------------------<assignment_statment>
518
519 ----------------------<id>
520
521 -----------------------<char>
522
523 ------------------------[a]
524
525 ---------------------[=]
526
527 ---------------------<expr>
528
529 ----------------------<int_expr>
530
531 -----------------------<digit>
532
533 ------------------------[1]
534
535 -----------------------<intop>
536
537 ------------------------[+]
538
539 -----------------------<expr>
540
541 ------------------------<int_expr>
542
```

```
543  ------------------------<digit>
544
545  -------------------------[2]
546
547  -------------------<statement_list>
548
549  -------------------<statement>
550
551  --------------------<block>
552
553  ---------------------[{]
554
555  ----------------------<statement_list>
556
557  -----------------------<statement>
558
559  ------------------------<print_statment>
560
561  -------------------------[print]
562
563  -------------------------[(]
564
565  -------------------------<expr>
566
567  --------------------------<id>
568
569  ---------------------------<char>
570
571  ----------------------------[b]
572
573  -------------------------[)]
574
575  ----------------------[statement_list]
576
577  ---------------------[}]
578
579  -------------------[statement_list]
580
581  -----------------[}]
582
583  ----------------<statement_list>
584
585  -----------------<statement>
586
587  ------------------<assignment_statment>
588
589  ------------------<id>
590
591  -------------------<char>
592
593  --------------------[b]
594
595  -------------------[=]
596
```

```
------------------<expr>
--------------------<id>
--------------------<char>
---------------------[a]
-----------------<statement_list>
------------------<statement>
------------------<print_statment>
-------------------[print]
-------------------[(]
-------------------<expr>
---------------------<id>
---------------------<char>
----------------------[b]
-------------------[)]
------------------[statement_list]
----------[}]
--------<statement_list>
---------<statement>
----------<print_statment>
-----------[print]
-----------[(]
-----------<expr>
------------<id>
-------------<char>
--------------[b]
-----------[)]
---------[statement_list]
```

```
651    --[}]

652

653    -[$]

654

655    End of program 5
```

1.6 Test Case 6

```
1    /*invalid 6*/
2    }$
3
4    Paser starting :)
5    Parseing: Parse_Program()
6    Parseing: Parse_Block()
7    Parse Error: Expected { but found } at 76 : 1
8    Parseing: Parse_Statement_List()
9    Parse for Program 6 Done
10   Parse has 1 errors :(
11   No CST will be made :(
12   End of program 6
```

1.7 Test Case 7

```
1    /*valid 7*/
2    {
3
4            int a
5            a = 3
6            int b
7            b = 4
8            a = b
9            print(a)
10           if (a == b) {
11               print(a)
12           }
13       }$
14
15   Paser starting :)
16   Parseing: Parse_Program()
17   Parseing: Parse_Block()
18   Parseing: Parse_Statement_List()
19   Parseing: Parse_Statement()
20   Parseing: Parse_Var_Decl()
21   Parseing: Parse_Type()
22   Parseing: Parse_Id()
23   Parseing: Parse_Char()
24   Parseing: Parse_Statement_List()
25   Parseing: Parse_Statement()
26   Parseing: Parse_Assignment_Statment()
27   Parseing: Parse_Id()
28   Parseing: Parse_Char()
29   Parseing: Parse_Expr()
```

```
30  Parseing: Parse_Int_Expr()
31  Parseing: Parse_Digit()
32  Parseing: Parse_Statement_List()
33  Parseing: Parse_Statement()
34  Parseing: Parse_Var_Decl()
35  Parseing: Parse_Type()
36  Parseing: Parse_Id()
37  Parseing: Parse_Char()
38  Parseing: Parse_Statement_List()
39  Parseing: Parse_Statement()
40  Parseing: Parse_Assignment_Statment()
41  Parseing: Parse_Id()
42  Parseing: Parse_Char()
43  Parseing: Parse_Expr()
44  Parseing: Parse_Int_Expr()
45  Parseing: Parse_Digit()
46  Parseing: Parse_Statement_List()
47  Parseing: Parse_Statement()
48  Parseing: Parse_Assignment_Statment()
49  Parseing: Parse_Id()
50  Parseing: Parse_Char()
51  Parseing: Parse_Expr()
52  Parseing: Parse_Id()
53  Parseing: Parse_Char()
54  Parseing: Parse_Statement_List()
55  Parseing: Parse_Statement()
56  Parseing: Parse_Print_Statment()
57  Parseing: Parse_Expr()
58  Parseing: Parse_Id()
59  Parseing: Parse_Char()
60  Parseing: Parse_Statement_List()
61  Parseing: Parse_Statement()
62  Parseing: Parse_If_Statment()
63  Parseing: Parse_Boolean_Expr()
64  Parseing: Parse_Expr()
65  Parseing: Parse_Id()
66  Parseing: Parse_Char()
67  Parseing: Parse_Boolop()
68  Parseing: Parse_Expr()
69  Parseing: Parse_Id()
70  Parseing: Parse_Char()
71  Parseing: Parse_Block()
72  Parseing: Parse_Statement_List()
73  Parseing: Parse_Statement()
74  Parseing: Parse_Print_Statment()
75  Parseing: Parse_Expr()
76  Parseing: Parse_Id()
77  Parseing: Parse_Char()
78  Parseing: Parse_Statement_List()
79  Parseing: Parse_Statement_List()
80  Parse for Program 7 Done
81  Parse No errors :)
82  CST will be displayed :)
83  <program>
```

```
  84
  85  -<block>
  86
  87  --[{]
  88
  89  --<statement_list>
  90
  91  ---<statement>
  92
  93  ----<var_decl>
  94
  95  -----<type>
  96
  97  ------[int]
  98
  99  -----<id>
 100
 101  ------<char>
 102
 103  -------[a]
 104
 105  ---<statement_list>
 106
 107  ----<statement>
 108
 109  -----<assignment_statment>
 110
 111  ------<id>
 112
 113  -------<char>
 114
 115  --------[a]
 116
 117  ------[=]
 118
 119  ------<expr>
 120
 121  -------<int_expr>
 122
 123  --------<digit>
 124
 125  ---------[3]
 126
 127  ----<statement_list>
 128
 129  -----<statement>
 130
 131  ------<var_decl>
 132
 133  -------<type>
 134
 135  --------[int]
 136
 137  -------<id>
```

```
138
139    --------<char>
140
141    ---------[b]
142
143    -----<statement_list>
144
145    ------<statement>
146
147    -------<assignment_statment>
148
149    --------<id>
150
151    ---------<char>
152
153    ----------[b]
154
155    --------[=]
156
157    --------<expr>
158
159    ---------<int_expr>
160
161    ----------<digit>
162
163    -----------[4]
164
165    ------<statement_list>
166
167    -------<statement>
168
169    --------<assignment_statment>
170
171    ---------<id>
172
173    ----------<char>
174
175    -----------[a]
176
177    ---------[=]
178
179    ---------<expr>
180
181    ----------<id>
182
183    -----------<char>
184
185    ------------[b]
186
187    -------<statement_list>
188
189    --------<statement>
190
191    ---------<print_statment>
```

```
192
193    ----------[print]
194
195    ----------[(]
196
197    ----------<expr>
198
199    -----------<id>
200
201    ------------<char>
202
203    ------------[a]
204
205    ----------[)]
206
207    --------<statement_list>
208
209    ---------<statement>
210
211    ----------<if_statment>
212
213    -----------[if]
214
215    -----------<boolean_expr>
216
217    ------------[(]
218
219    ------------<expr>
220
221    -------------<id>
222
223    --------------<char>
224
225    --------------[a]
226
227    ------------<boolop>
228
229    -------------[==]
230
231    ------------<expr>
232
233    -------------<id>
234
235    --------------<char>
236
237    --------------[b]
238
239    ------------[)]
240
241    -----------<block>
242
243    ------------[{]
244
245    ------------<statement_list>
```

```
246
247    -------------<statement>
248
249    -------------<print_statment>
250
251    --------------[print]
252
253    --------------[(]
254
255    --------------<expr>
256
257    ---------------<id>
258
259    ----------------<char>
260
261    -----------------[a]
262
263    --------------[)]
264
265    ------------[statement_list]
266
267    -----------[}]
268
269    ---------[statement_list]
270
271    --[}]
272
273    -[$]
274
275    End of program 7
```

1.8   Test Case 8

```
1    /*invalid 8*/
2    {
3        if(!=)
4    }$
5
6    Paser starting :)
7    Parseing: Parse_Program()
8    Parseing: Parse_Block()
9    Parseing: Parse_Statement_List()
10   Parseing: Parse_Statement()
11   Parseing: Parse_If_Statment()
12   Parseing: Parse_Boolean_Expr()
13   Parseing: Parse_Expr()
14   Parse Error: Expected [IntExpr, StringExpr, BooleanExpr, ID] but found != at 94 : 9
15   Parseing: Parse_Boolop()
16   Parseing: Parse_Expr()
17   Parse Error: Expected [IntExpr, StringExpr, BooleanExpr, ID] but found ) at 94 : 10
18   Parseing: Parse_Block()
19   Parse Error: Expected { but found } at 95 : 1
20   Parseing: Parse_Statement_List()
```

```
21  Parseing: Parse_Statement_List()
22  Parse Error: Expected } but found $ at 95 : 2
23  Parse for Program 8 Done
24  Parse has 4 errors :(
25  No CST will be made :(
26  End of program 8
```

1.9   Test Case 9

```
1  /*invalid 9*/
2  {
3      print(7 print a)
4  }$
5
6  Paser starting :)
7  Parseing: Parse_Program()
8  Parseing: Parse_Block()
9  Parseing: Parse_Statement_List()
10  Parseing: Parse_Statement()
11  Parseing: Parse_Print_Statment()
12  Parseing: Parse_Expr()
13  Parseing: Parse_Int_Expr()
14  Parseing: Parse_Digit()
15  Parse Error: Expected ) but found print at 99 : 17
16  Parseing: Parse_Statement_List()
17  Parseing: Parse_Statement()
18  Parseing: Parse_Print_Statment()
19  Parse Error: Expected ( but found a at 99 : 19
20  Parseing: Parse_Expr()
21  Parseing: Parse_Id()
22  Parseing: Parse_Char()
23  Parseing: Parse_Statement_List()
24  Parse for Program 9 Done
25  Parse has 2 errors :(
26  No CST will be made :(
27  End of program 9
```

1.10   Test Case 10

```
1  /*valid 10*/
2  {
3      print(7)
4  }$
5
6  Paser starting :)
7  Parseing: Parse_Program()
8  Parseing: Parse_Block()
9  Parseing: Parse_Statement_List()
10  Parseing: Parse_Statement()
11  Parseing: Parse_Print_Statment()
12  Parseing: Parse_Expr()
13  Parseing: Parse_Int_Expr()
```

```
14  Parseing: Parse_Digit ()
15  Parseing: Parse_Statement_List ()
16  Parse for Program 10 Done
17  Parse No errors :)
18  CST will be displayed :)
19  <program >
20
21  -<block >
22
23  --[{]
24
25  --<statement_list >
26
27  ---<statement >
28
29  ----<print_statment >
30
31  -----[print]
32
33  -----[(]
34
35  -----<expr >
36
37  ------<int_expr >
38
39  -------<digit >
40
41  --------[7]
42
43  -----[)]
44
45  ---[statement_list]
46
47  --[}]
48
49  -[$]
50
51  End of program 10
```

1.11 Test Case 11

```
1  /*valid 11*/
2  {
3      print (1+2+3+4)
4  }$
5
6  Paser starting :)
7  Parseing: Parse_Program ()
8  Parseing: Parse_Block ()
9  Parseing: Parse_Statement_List ()
10  Parseing: Parse_Statement ()
11  Parseing: Parse_Print_Statment ()
12  Parseing: Parse_Expr ()
```

```
13  Parseing: Parse_Int_Expr()
14  Parseing: Parse_Digit()
15  Parseing: Parse_intop()
16  Parseing: Parse_Expr()
17  Parseing: Parse_Int_Expr()
18  Parseing: Parse_Digit()
19  Parseing: Parse_intop()
20  Parseing: Parse_Expr()
21  Parseing: Parse_Int_Expr()
22  Parseing: Parse_Digit()
23  Parseing: Parse_intop()
24  Parseing: Parse_Expr()
25  Parseing: Parse_Int_Expr()
26  Parseing: Parse_Digit()
27  Parseing: Parse_Statement_List()
28  Parse for Program 11 Done
29  Parse No errors :)
30  CST will be displayed :)
31  <program>
32
33  -<block>
34
35  --[{]
36
37  --<statement_list>
38
39  ---<statement>
40
41  ----<print_statment>
42
43  -----[print]
44
45  -----[(]
46
47  -----<expr>
48
49  ------<int_expr>
50
51  -------<digit>
52
53  --------[1]
54
55  -------<intop>
56
57  --------[+]
58
59  -------<expr>
60
61  --------<int_expr>
62
63  ---------<digit>
64
65  ----------[2]
66
```

```
67    ---------<intop>

68

69    ----------[+]

70

71    ---------<expr>

72

73    ----------<int_expr>

74

75    -----------<digit>

76

77    -----------[3]

78

79    -----------<intop>

80

81    -----------[+]

82

83    -----------<expr>

84

85    ------------<int_expr>

86

87    -------------<digit>

88

89    --------------[4]

90

91    -----[)]

92

93    ---[statement_list]

94

95    --[}]

96

97    -[$]

98

99    End of program 11
```

2   References

Java Regular Expression
Enum
Switch cases (Was more of an idea)
Double Quote For Regular Expressions
Groups in Regular Expressions
More Group stuff
Index for Regular Expressions
Check for end of line
Full Syntax Java Regx
Quote In Regx

I used these all to help me understand how to use Regular Expression and get the data that I need in order to make the tokens. Enum and Cases where not used but more of an ideas that I was thing about using but never did end up using them. No changes to the references, technically I used the https://www.labouseur.com/index.html website for the guided code for the parser and notes from the slides describing how the parser should act.