



컴파일러

전이다이어그램을 이용한 어휘분석기 과제 #2

제출일 2022.04.21

학과 소프트웨어학과

학번 2020039009

이름 차현아

담당교수 이재성 교수님



실행 결과 - 과제 파일 예시

Microsoft Visual Studio 디버그 콘솔

입력 :for\$
1
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :if\$
2
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :int\$
3
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :f1\$
9
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :i2x\$
9
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :ifs\$
9
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :if(\$
2 invalid char '('
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :B123\$
9
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :intger\$
9
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :7\$
invalid char '7'
C:\Users\User\source\repos\Project21\Project21\bin\Debug\Project21.exe
이 창을 닫으려면 아무 키나 누르세요...

실행 결과 - 추가 예시

Microsoft Visual Studio 디버그 콘솔

입력 :(((
 invalid char '(' invalid char '(' invalid char '('
 C:\Users\User\source\repos\Project21\Debug\Project21.exe
 이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :fo(\$
 9 invalid char '('
 C:\Users\User\source\repos\Project21\Debug\Project21.exe
 이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :345love12\$
 invalid char '3' invalid char '4' invalid char '5' 9
 C:\Users\User\source\repos\Project21\Debug\Project21.exe(프
 이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :in2(\$
 9 invalid char '('
 C:\Users\User\source\repos\Project21\Debug\Project21.exe
 이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :for(int))\$
 1 invalid char '(' 3 invalid char ')' invalid char ')'
 C:\Users\User\source\repos\Project21\Debug\Project21.exe(프
 이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :i(((
 9 invalid char '(' invalid char '(' invalid char '('
 C:\Users\User\source\repos\Project21\Debug\Project21.exe
 이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :int#for(\$
 3 invalid char '#' 1 invalid char '('
 C:\Users\User\source\repos\Project21\Debug\Project21.exe
 이 창을 닫으려면 아무 키나 누르세요...

Microsoft Visual Studio 디버그 콘솔

입력 :int((if)\$
 3 invalid char '(' invalid char '(' 2 invalid char ')'
 C:\Users\User\source\repos\Project21\Debug\Project21.exe(프
 이 창을 닫으려면 아무 키나 누르세요...

코드

```

1  #include<stdio.h>
2  #include<ctype.h>
3
4  // (알파벳) or (알파벳 + 숫자) 토큰을 처리하기 위한 전역변수
5  int count = 0;
6
7  int state = 0;
8  char nextchar();
9  int nexttoken();
10 char c;
11 void retraction(char c); // 입력 되돌리기

```

- (문자) or (문자+숫자) 토큰을 리턴하기 위해서 전역변수 count를 사용하였음.

ex) fo에서 뒷문자가 r이 아니라 그 외의 값(문자,숫자,invalid)이면 9를 리턴해야함. 뒤쪽에 invalid값이 있으면 9를 리턴한 뒤, invalid를 출력해야함.

코드

```

13 void main()
14 {
15     int num;
16     printf("입력 :");
17     while (1)
18     {
19         num = nexttoken();
20         count = 0;
21         if (num != -1) // c가 입력의 끝을 나타내는 문자인 '$'라면, -1을 리턴
22         {
23             printf("%d ", num); // 리턴 값 출력
24         }
25
26         if (c == '$') // c가 문자입력 종료를 나타내는 기호이면 종료
27             break;
28         else // 그렇지않으면 모두 invalid값이므로 리트랙션 한뒤, state = 11
29         {
30             retraction(c);
31             state = 11;
32         }
33     }
34     return;
35 }

```

- 입력의 끝을 나타내는 기호는 '\$'로 설정하였음.

- nexttoken()에서는 입력 문자열을 한 문자씩 읽어들이고 토큰을 만나면 토큰처리 값을 리턴함. 리턴값이 -1이라면 출력하지않음(state 0에서 c가 '\$'일 경우 -1을 리턴하도록 설계)

- nexttoken()에서 nextchar()의 getchar()을 통해 읽어들이는 값 c가 '\$'이 아닌 경우는 모두 invalid한 값이 도출되므로 리트랙션을 한 뒤 state 11로 가도록 만들었음.

```

37 char nextchar()
38 {
39     while (1)
40     {
41         c = getchar();
42         return c;
43     }
44 }

```

코드

```

46 int nexttoken()
47 {
48     while (1)
49     {
50         switch (state)
51         {
52             case 0:
53                 c = nextchar();
54                 if (c == '+') // c가 +면 state=1, count값 증가
55                 {
56                     state = 1;
57                     count++; // 9를 리턴할 수 있도록 만듦
58                 }
59                 else if (c == 'i') // c가 i면 state=5
60                 {
61                     state = 5;
62                     count++;
63                 }
64                 else if (isalpha(c)) // c가 i도 아니고 f도 아닌 알파벳이라면 state=11
65                 {
66                     state = 11;
67                     count++;
68                 }
69                 else if (c == '$') // c가 입력의 끝을 나타내는 문자인 '$'라면, -1을 리턴
70                     return -1;
71                 else // 위 조건을 모두 만족하지 않으면 state=12
72                     state = 12;
73                 break;
74             case 1:
75                 c = nextchar();
76                 if (c == 'o')
77                     state = 2;
78                 else if (isalnum(c)) // 문자나 숫자라면 state = 11
79                 {
80                     state = 11;
81                 }
82                 else // invalid 값이라면 state = 12
83                 {
84                     state = 12;
85                 }
86                 break;
87             case 2:
88                 c = nextchar();
89                 if (c == 'r')
90                     state = 3;
91                 else if (isalnum(c)) // 문자나 숫자라면 state = 11
92                 {
93                     state = 11;
94                 }
95                 else
96                 {
97                     state = 12;
98                 }
99                 break;
100             case 3:
101                 c = nextchar();
102                 if (isalnum(c)) // 문자나 숫자라면 state = 11
103                     state = 11;
104                 else // 그 외의 값이라면, state = 4로 가서 1리턴(토큰 for 인식)
105                     state = 4;
106                 break;
107             case 4:
108                 return 1;
109                 break;
110             case 5:
111                 c = nextchar();
112                 if (c == 'f')
113                     state = 6;
114                 else if (c == 'n')
115                     state = 8;
116                 else if (isalnum(c)) // 문자나 숫자라면 state = 11
117                 {
118                     state = 11;
119                 }
120                 else // 그렇지 않으면 state = 12 로 가서 9 리턴
121                 {
122                     state = 12;
123                 }
124                 break;

```

코드

```

125 case 6:
126     c = nextchar();
127     if (isalnum(c))
128         state = 11;
129     else
130         state = 7;
131     break;
132 case 7:
133     return 2;
134     break;
135 case 8:
136     c = nextchar();
137     if (c == 't')
138         state = 9;
139     else if (isalnum(c))
140     {
141         state = 11;
142     }
143     else
144     {
145         state = 12;
146     }
147     break;
148 case 9:
149     c = nextchar();
150     if (isalnum(c))
151         state = 11;
152     else
153         state = 10;
154     break;
155 case 10:
156     return 3;
157     break;
158 case 11:
159     c = nextchar();
160     if (isalnum(c)) // 문자나 숫자라면 다시 state = 1
161     {
162         count++;
163         state = 11;
164     }
165     else // 리트랙션 한 뒤의 invalid값이라면 state = 12
166         state = 12;
167     break;
168 case 12:
169     if (count > 0) // 알파벳 또는 알파벳+ 숫자 조합이면 토큰이므로 9리턴
170         return 9;
171
172     if (c != '$') // 문자 입력 종료가 아니면 모두 invalid값이므로 오류 출력
173     {
174         printf("invalid char '%c' ", c);
175         state = 0; // state = 0 으로 가서 남아있는 입력값 처리
176     }
177     break;
178 }
179 }
180 }
181
182 void retraction(char c) {
183     ungetc(c, stdin);
184 }

```

- state 1이나 5로 진입한 뒤, 토큰 값이 아닌 문자나 숫자가 나오면 state 11로 가서 작업을 수행.
- state 1이나 5로 진입한 뒤, 문자나 숫자가 아닌 값이 나오면 일단 9를 리턴해야하므로 state 12로 가서 9를 리턴한 뒤 현재 읽은 문자가 '\$'라면 작업 종료, 그렇지 않으면 리트랙션 수행 후 state 11로 이동하도록 설계하였음.
- for이나 if, int 토큰이 인식되면, 일단 토큰에 해당하는 값을 리턴해준 뒤 현재 읽은 문자가 \$라면 작업 종료, 그렇지 않으면 invalid값이므로 리트랙션 수행 후 state 11로 이동.
- 입력값의 시작은 숫자가 될 수 없으므로 state 0에서 처음으로 읽은 값이 숫자라면, state 12로 이동해서 invalid값을 출력

전이 다이어그램

2020039009- 차현아

