Ahsanullah University of Science & Technology

Department of Computer Science & Engineering

Course No. : CSE 4130

Course Name : Formal Languages and Compilers Lab

Assignment No. : 04

Submitted By:

Name : Mohammad Shah Alam

ID No. : 17-01-04-012

Session : Spring - 2020

Section : A (A1)

QUESTION :

Suppose, a given C source program has been scanned, filtered, lexically analyzed and tokenized as that were done in earlier sessions. In addition, line numbers have been assigned to the source code lines for generating proper error messages. As the first step to Syntax Analysis, we now perform detection of simple syntax errors like duplication of tokens except parentheses or braces, unbalanced braces or parentheses problem, unmatched 'else' problem, etc. Duplicate identifier declarations must also be detected with the help of the Symbol Table.

ANSWER:

```
#include<bits/stdc++.h>
using namespace std;
FILE *file1,*file2;
char ch,ch1,temp,temp1;

void funcRemovingExtraSpaceComments()
{
    file1=fopen("input.cpp","r");
    file2=fopen("output1.cpp","w");

    if(!file1)
        cout<<"File can't be opened"<<endl;
    else
    {
        while((ch=fgetc(file1))!=EOF)
        {
            if(ch=='/')
            {
                ch=fgetc(file1);

                if(ch=='/')
                {
                    while(ch!='\n')
```

```
                {
                    ch=fgetc(file1);
                }
            }

            else if(ch=='*')
            {
                while((ch=fgetc(file1))!=EOF)
                {
                    if(ch=='*')
                    {
                        ch=fgetc(file1);
                        if(ch=='/')
                        {
                            break;
                        }
                    }
                }
            }
            else
            {
                fputc('/',file2);
                fputc(ch,file2);
            }
        }
        else if(ch == '\n')
        {
            fputc('\n',file2);
        }
        else if(ch == ' ')
        {
            while(ch ==' ' || ch == '\n' )
            {

                if(ch == '\n')
                {
```

```
                    fputc('\n',file2);
                    break;
                }
                ch = fgetc(file1);

            }
            fputc(' ',file2);
            fputc(ch,file2);
        }
        else
        {
            fputc(ch,file2);
        }
    }
    fclose(file1);
    fclose(file2);


    cout<<endl<<"After removing spaces and comment the output is :"<<endl;

    file2=fopen("output1.cpp","r");
    while((ch=fgetc(file2))!=EOF)
    {
        cout<<ch;
    }
    fclose(file2);
}


void funcAddingLineNumber()
{
    int countLine = 1;
    file1 = fopen("output1.cpp","r");
    file2 = fopen("output2.cpp","w");
```

```
    if(!file1)
        cout<<"File can't be opened"<<endl;
    else
    {
        fprintf(file2,"%d",countLine);
        while((ch=fgetc(file1))!=EOF)
        {
            if(ch!='\n')
                fputc(ch,file2);
            else
            {
                countLine++;
                fputc(ch,file2);
                fprintf(file2,"%d",countLine);
            }
        }
    }
    fclose(file1);
    fclose(file2);

    cout<<endl<<"After second step done with line number the output is :"<<endl;
    file2=fopen("output2.cpp","r");
    while((ch=fgetc(file2))!=EOF)
    {
        cout<<ch;
    }
    fclose(file2);
}

void funcIdToken()
{
    file1=fopen("output2.cpp","r");
    file2=fopen("output3.cpp","w");
    if(!file1)
    {
        cout<<"File not found"<<endl;
```

```
    }
    else
    {
        while((ch=fgetc(file1))!=EOF)
        {
            //check for float
            if(ch=='f')
            {
                ch=fgetc(file1);
                if(ch=='l')
                {
                    ch=fgetc(file1);
                    if(ch=='o')
                    {
                        ch=fgetc(file1);
                        if(ch=='a')
                        {
                            ch=fgetc(file1);
                            if(ch=='t')
                            {
                                fputs("float",file2);
                            }
                            else
                            {
                                fputs("id floa",file2);
                            }
                        }
                        else
                        {
                            fputs("id flo",file2);
                        }
                    }
                    else
                    {
                        fputs("id fl",file2);
                    }
```

```
            }
            else if (isdigit(ch))
            {
                fputs("id ",file2);
                fputc('f',file2);
                fputc(ch,file2);
            }
            else
            {
                fputs("id f",file2);
            }
        }
        //check for int
        else if(ch=='i')
        {
            ch=fgetc(file1);
            if(ch=='n')
            {
                ch=fgetc(file1);
                if(ch=='t')
                {
                    fputs("int",file2);

                }
                else
                {
                    fputs("id in",file2);
                }
            }
            //check for if
            else if(ch=='f')
            {
                fputs("if",file2);
            }
            else
            {
```

```
                fputs("id i",file2);
            }
        }
        //check for char
        else if(ch=='c')
        {
            ch=fgetc(file1);
            if(ch=='h')
            {
                ch=fgetc(file1);
                if(ch=='a')
                {
                    ch=fgetc(file1);
                    if(ch=='r')
                    {
                        fputs("char",file2);

                    }
                    else
                    {
                        fputs("id cha",file2);
                    }
                }
                else
                {
                    fputs("id ch",file2);
                }
            }
            else
            {
                fputs("id c",file2);
            }
        }

        //check for else
        else if(ch=='e')
```

```
        {
            ch=fgetc(file1);
            if(ch=='l')
            {
                ch=fgetc(file1);
                if(ch=='s')
                {
                    ch=fgetc(file1);
                    if(ch=='e')
                    {
                        fputs("else",file2);
                    }
                    else
                    {
                        fputs("id els",file2);
                    }
                }
                else
                {
                    fputs("id el",file2);
                }
            }
            else
            {
                fputs("id e",file2);
            }
        }
        //check for main
        else if (ch=='m')
        {
            ch1 = ch;
            ch=fgetc(file1);
            if(ch=='a')
            {
                ch=fgetc(file1);
                ch=fgetc(file1);
```

```
                fputs("id main",file2);
            }
        }
        //check for void
        else if (ch=='v')
        {
            ch1=ch;
            ch=fgetc(file1);
            if(ch=='o')
            {
                ch=fgetc(file1);
                ch=fgetc(file1);
            }
            fputs("void",file2);
        }
        //check for return
        else if (ch=='r')
        {
            ch1=ch;
            ch=fgetc(file1);
            if(ch=='e')
            {
                ch=fgetc(file1);
                ch=fgetc(file1);
                ch=fgetc(file1);
                ch=fgetc(file1);
                fputs("return",file2);
            }
        }
        //check for double
        else if (ch=='d')
        {
            ch1=ch;
            ch=fgetc(file1);
            if(ch=='o')
            {
```

```
                ch1=ch;
                ch=fgetc(file1);
                if(ch=='u')
                {
                    ch=fgetc(file1);
                    ch=fgetc(file1);
                    ch=fgetc(file1);
                    fputs("double",file2);
                }

            }
        }
        //check for identifier
        else if(isalpha(ch))
        {
            temp=ch;
            ch=fgetc(file1);
            if(isdigit(ch))
            {
                fputs("id ",file2);
                fputc(temp,file2);
                fputc(ch,file2);
            }
            else if(ch=='_')
            {
                temp1=ch;
                ch=fgetc(file1);
                fputs("id ",file2);
                fputc(temp,file2);
                fputc(temp1,file2);
                fputc(ch,file2);
            }
            else
            {
                fputs("id ",file2);
                fputc(temp,file2);
```

```
                    fputc(ch,file2);
                }
            }
            //else
            else
            {
                fputc(ch,file2);
            }

        }
    }

    fclose(file1);
    fclose(file2);

    cout<<endl<<"After third step done the output is:"<<endl;
    file2=fopen("output3.cpp","r");
    while((ch=fgetc(file2))!=EOF)
    {
        cout<<ch;
    }
    fclose(file2);

}

void funcError()
{
    int line1 = 1,line2 = 1,ifCounter = 0,elseCounter = 0,bracOpenCount = 0,bracCloseCount
= 0;
    cout<<endl<<"After last step done:"<<endl;
    file1=fopen("output2.cpp","r");
    if(!file1)
        cout<<"File can't be opened"<<endl;
    else
    {
        while((ch=fgetc(file1))!=EOF)
```

```
    {
        if(ch=='i')
        {
            ch=fgetc(file1);
            if(ch=='f')
            {
                ifCounter++;
            }
        }
        else if(ch=='e')
        {
            ch=fgetc(file1);
            if(ch=='l')
            {
                ch=fgetc(file1);
                if(ch=='s')
                {
                    ch=fgetc(file1);
                    if(ch=='e')
                    {
                        if(ifCounter<=0)
                        {
                            cout<<"Unmatched 'else' at line"<<line1<<endl;
                        }
                        else
                        {
                            elseCounter++;
                            if(elseCounter>ifCounter)
                            {
                                cout<<"Unmatched 'else' at line"<<line1<<endl;
                            }
                            else
                            {
                                ifCounter--;
                            }
```

```
                    }
                }
            }

        }
    }

        else if(ch=='{')
    {
        bracOpenCount++;
    }
        else if(ch=='\n')
    {
        line1++;
    }
        else if(ch=='}')
    {
        if(bracOpenCount<=0)
        {
            cout<<"Unmatched '}' at line "<<line1<<endl;
        }
        else
        {
            bracCloseCount++;
            if(bracCloseCount>bracOpenCount)
            {
                cout<<"Unmatched '}' at line "<<line1<<endl;
            }
            else if(bracOpenCount>bracCloseCount)
            {
                cout<<"Unmatched '{' at line "<<line1<<endl;
            }
            else
            {
                bracOpenCount--;
            }
```

```
            }
        }

    }

  }
  fclose(file1);

  file1=fopen("output2.cpp","r");
  if(!file1)
     cout<<"File can't be opened"<<endl;
  else
   {
      while((ch=fgetc(file1))!=EOF)
      {
          if(ch=='f')
          {
              ch=fgetc(file1);
              if(ch=='o')
              {
                  ch=fgetc(file1);
                  if(ch=='r')
                  {
                      while((ch=fgetc(file1))!=')')
                      {

                          if(ch==';')
                          {
                              ch=fgetc(file1);
                              if(ch==';')
                              {
                                  cout<<"Duplicate token at line "<<line2<<endl;
                              }

                          }
```

```
                    }
                }
            }
        }
        else if(ch==';')
        {
            ch=fgetc(file1);
            if(ch==';')
            {
                cout<<"Duplicate token at line "<<line2<<endl;
            }

        }
        else if(ch=='\n')
        {
            // cout<< " line 2 value is  " <<line2 <<" "<< endl;
            line2++;
        }
    }
  }
  fclose(file1);
}


int main()
{
  char choose;
  while(choose != 't')
  {
      cout<<endl<<"Enter 1 for for removing extra spaces and comments "<<endl<<"Enter
2 for adding line numbers"<<endl<<"Enter 3 for tokenizing "<<endl<<"Enter 4 for error
check "<<endl<<"Enter t for terminate"<<endl<<endl;
      cin>>choose;
      switch(choose)
      {
      case '1':
```

```
            funcRemovingExtraSpaceComments();
            break;
        case '2':
            funcAddingLineNumber();
            break;
        case '3':
            funcIdToken();
            break;
        case '4':
            funcError();
            break;

        case 't':
            exit(0);
        }


    }


    return 0;
}
```