

Práctica Flex

Trabajo realizado por: Antonio Haro Arriaza

Problema original establecido

Se tiene un archivo de texto que es dado como argumento al ejecutar el archivo. Este archivo de texto suponemos que contiene información de clientes donde han pasado un filtro y se tiene como mínimo un número de teléfono fijo, teléfono móvil o un correo (solo admite @gmail.com, @hotmail.es, @hotmail.com, @correo.ugr.es) y suponemos que sí tienen ambas cosas están seguidas y separadas por un "-". Este programa reconoce estos conjuntos y los guarda en otro archivo para poder usarlo como lista resumida para poder hacer luego una lista de spam más tarde o simplemente tenerlo más ordenado. En el programa lo he enfocado como una lista más ordenada de datos.

Empezando por la parte más perteneciente a MC:

he realizado tres declaraciones:

```
%}  
%option c++  
numero (6[0-9]{8})|(9[0-9]{8})  
correo (.*)(@gmail.com|@hotmail.es|@hotmail.com|@correo.ugr.es)  
numcorreo ({numero}"-"{correo})|({correo}"-"{numero})  
%%  
{numcorreo} { return 1; }  
{numero} { return 2; }  
{correo} { return 3; }  
%%
```

número -> empieza por 6 y luego tiene una cadena desde 0 hasta 9 de una longitud de 8 caracteres, el símbolo "|" que es como una unión que elige (6[0-9]{8}) o (9[0-9]{8}) y lo mismo pero que empieza por 9 para fijos.

correo -> (.)* que significa cualquier carácter repetido infinitamente exceptuando un salto de línea seguido de una unión concatenada -> (@gmail.com|@hotmail.es|@hotmail.com|@correo.ugr.es) donde solo se permite una de estas.

numcorreo -> teniendo ya en cuenta que tenemos declarado correo y numero simplemente hice una unión de estas concatenadas con un “-” que significa el carácter literal situado entre las comillas ,también podría ponerse de forma manual poniendo donde número su declaración pero como flex me lo daba por bueno al ver que cuando ponía el cursor me salía la declaración antes dada supuse que estaba bien.

En las Reglas simplemente puse 3 y cada una retornará un número distinto en la función yylex();

Programa C++

```
%{  
    #include <fstream>  
    #include <iostream>  
    #include <vector>  
    #include <FlexLexer.h>  
    #include <utility>  
    using namespace std;  
    ifstream fichero;  
    ofstream Esc;  
    vector<pair<const char*,string> > vecclientes;  
    const char* first;  
    string second;  
    pair<string,string> cli;  
    void aniadir(vector<pair<const char*,string> > vec);  
}%
```

Las bibliotecas usadas en este programa son fstream para el manejo de ficheros e iostream, vector y utility para poder usar vector de pair donde guardare la información del cliente por eso en pair<> hay un const char* porque en los apuntes donde se guarda los tokens(el texto reconocido por las reglas y declaraciones) es en *yyltext* que es un const char*.

ifstream para lectura y ofstream para escritura .Las demás declaraciones son muy simples y también la declaración de la función aniadir que la uso como escritura ya que vi en los apuntes que se declara todo ahí en ese apartado del principio.

```
void aniadir(vector<pair<const char*,string> > vec){
    for(int i = 0;i < vec.size(); i++){
        Esc << "Cliente "<< i << ": " << vec[i].first << " " << vec[i].second << endl ;
    }
}
```

La primera función es simplemente para escribir en el archivo. Lo siguiente del programa es una main donde se abren los archivos y se usan las funciones de la biblioteca FlexLexer.h un switch para meter los siguientes datos dependiendo de que se obtenga un número ,un correo o ambos.

Compilacion

-flex Practica1.l

```
C:\Users\antonio\Desktop\UNIVERSIDAD\Proyectos\C++(flex)>g++ lex.yy.cc -I \Users\antonio\AppData\Local\Packages\CanonicalGroupLimited.Ubuntu18.04onWindows_79r
p1fndgsc\LocalState\rootfs\usr\include\
```

g++ lex.yy.cc -I {path de la libreria FlexLexer.h}

En la compilación me dan errores de la librería FlexLexer.h aun declarándolas bien, en la única función es en la de

- yyFlexLexer(istream* arg_yyin = 0, ostream* arg_yyout = 0) construye un objeto yyFlexLexer usando los flujos dados para la entrada y salida. Si no se especifica, los flujos se establecen por defecto a cin y cout, respectivamente.

donde leí por internet que si es de lectura es en istream -> & fichero y 0 en la otra para null y para escritura en ostream -> & fichero y 0 en istream para null .Me salen muchos errores relacionados a esta función en el programa transformado en C sobre todo de matching error .Estuve mucho tiempo intentando distintas formas de declararlas según personas en stackoverflow pero nada funcionó.