

Αρχές Γλωσσών Προγραμματισμού και Μεταφραστών

Εργαστηριακή Άσκηση

Μέλη ομάδας:

Ονοματεπώνυμο: Κορνέλιου Ζούζα
AM: 236298

Ονοματεπώνυμο: Πέτρος Τανάκας
AM: 236361

Ονοματεπώνυμο: Γάτου Παρασκευή
AM: 1054370

Ονοματεπώνυμο: Γεράσιμος Χαράλαμπος Κόμης
AM: 1046076

Το BNF είναι:

```
<json> := '{' body '}' ;
<body>:= <parts> | <body> ',' <parts>;
<parts>:= <text> | <id_str> | <user> |
          <created_at> | <other> | <retweeted_status> | <truncated>
          | <extended_tweet> | <tweet>;
<created_at>:=      JSON_CREATEDAT ':' JSON_DAY JSON_MONTH JSON_NUM
JSON_NUM ':' JSON_NUM ':' JSON_NUM JSON_NUM ;
<user>:= JSON_USER ':' '{' <userbody> '}' ;
<id_str>:= JSON_IDSTR ':' JSON_STRNUM ;
<text>:= JSON_TEXT ':' JSON_STRING ;
<other>:= JSON_STRING ':' JSON_STRING ;
<userbody>:= <user_parts> | <userbody> ',' <user_parts>;
<user_parts>:=      <id> | <name> | <screen_name> | <location> | <other>;
<id>:=      JSON_ID ':' JSON_NUM;
<name>:=     JSON_NAME ':' JSON_STRING;
<screen_name>:= JSON_SCREENNAME ':' JSON_STRING;
<location>:=  JSON_LOCATION ':' JSON_STRING;
<retweeted_status>:=      JSON_RETWEETEDSTATUS ':' '{' rtbody '}' ;
<rtbody>:=    <text> ',' <user> | <user> ',' <text>;
<tweet>:=     JSON_TWEET ':' '{' <rtbody> '}' ;
<truncated>:=      JSON_TRUNCATED ':' TRUE ',' <display_text_range> |
JSON_TRUNCATED ':' FALSE;
<extended_tweet>:=  JSON_EXTENDED_TWEET '{' <full_text> ',' <display_text_range> '}' ;
<full_text> := JSON_FULLTEXT ':' STRING;
<display_text_range>:= JSON_DISPLAYTEXT_RANGE ':' <array_num>;
<array_num> : '[' <nums> ']';
<nums>: JSON_NUM | <nums> ',' JSON_NUM;
JSON_MONTH := Jan|Feb|Mar|Apr|May|Jun|Jul|Aug|Sep|Oct|Nov|Dec
JSON_DAY:= Mon|Thu|Wed|Tue|Fri|Sat|Sun
JSON_STRING:= \" Text \"
ALPHA := [a-zA-Z_]
DIGIT := [0-9]
```

```

DIGITS := {DIGIT}+
FRAC [.] {DIGITS}
EXP {E} {DIGITS}
E [eE] [+ -]?
HEX_DIGIT [0-9a-f]

UNESCAPEDCHAR [ -!#-[\]~]
ESCAPEDCHAR \\["\\bfnrt/]
UNICODECHAR \\u{HEX_DIGIT}{HEX_DIGIT}{HEX_DIGIT}{HEX_DIGIT}
CHAR {UNESCAPEDCHAR}|{ESCAPEDCHAR}|{UNICODECHAR}

```

Ο κώδικας του λεκτικού αναλυτή είναι:

```

% {
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include "code_parser.tab.h"
int line=1;
% }
ALPHA [a-zA-Z_]
DIGIT [0-9]
DIGITS {DIGIT}+
FRAC [.] {DIGITS}
EXP {E} {DIGITS}
E [eE] [+ -]?
HEX_DIGIT [0-9a-f]
UNESCAPEDCHAR [ -!#-[\]~]
ESCAPEDCHAR \\["\\bfnrt/]
UNICODECHAR \\u{HEX_DIGIT}{HEX_DIGIT}{HEX_DIGIT}{HEX_DIGIT}
CHAR {UNESCAPEDCHAR}|{ESCAPEDCHAR}|{UNICODECHAR}
STR {CHAR}+
%%

true                {printf("%s", yytext); return TRUE; }
false               {printf("%s", yytext); return FALSE; }
\"id\"              {printf("%s", yytext); return JSON_ID; }
\"id_str\"          {printf("%s", yytext); return JSON_IDSTR; }
\"text\"            {printf("%s", yytext); return JSON_TEXT; }
\"user\"            {printf("%s", yytext); return JSON_USER; }
\"name\"            {printf("%s", yytext); return JSON_NAME; }
\"screen_name\"     {printf("%s", yytext); return JSON_SCREENNAME; }
\"location\"        {printf("%s", yytext); return JSON_LOCATION; }
\"tweet\"           {printf("%s", yytext); return JSON_TWEET; }
\"retweeted_status\" {printf("%s", yytext); return JSON_RETWEETEDSTATUS; }
\"extended_tweet\"  {printf("%s", yytext); return JSON_EXTENDED_TWEET; }
\"created_at\"      {printf("%s", yytext); return JSON_CREATEDAT; }
Jan|Feb|Mar|Apr|May|Jun|Jul|Aug|Sep|Oct|Nov|Dec {printf("%s", yytext); return JSON_MONTH; }
Mon|Thu|Wed|Tue|Fri|Sat|Sun { printf("%s", yytext); return JSON_DAY; }
\{                {printf("{"); return '{';}
\}                {printf(")"); return '}';}
\"               {printf("\""); return JSON_QUOT; }
,                 {printf(","); return ','; }

```

```

:      {printf(":"); return ':';}
\"{DIGITS}\" {printf("%s", yytext); return JSON_STRNUM; }
\"{STR}\"      {printf("%s", yytext); return JSON_STRING; }
{DIGITS}      {printf("%s", yytext); return JSON_NUM; }
\n      {printf("%s",yytext); line++; }
[ ]      {printf("%s",yytext);};
[\t]      {printf("%s",yytext);};
%%

```

Ο κώδικας του συντακτικού αναλυτή είναι:

```

%{

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

FILE *yyin;

void yyerror(const char *w);

extern char* yytext;

extern int line;

int error=0;

%}

%token TRUE

%token FALSE

%token JSON_MONTH

%token JSON_DAY

%token JSON_IDSTR

%token JSON_STRNUM

%token JSON_USER

%token JSON_ID

%token JSON_TEXT

%token JSON_NAME

%token JSON_SCREENNAME

%token JSON_LOCATION

%token JSON_DISPLAYTEXTRANGE

%token JSON_CREATEDAT

```

%token JSON_RETWEETEDSTATUS

%token JSON_TWEET

%token JSON_TRUNCATED

%token JSON_QUOT

%token JSON_STRING

%token JSON_NUM

%token JSON_EXTENDED_TWEET

%token JSON_FULLTEXT

%%

json: '{' body '}';

body: parts | body ',' parts;

parts: text | id_str | user | created_at | other | retweeted_status | truncated | extended_tweet | tweet;

created_at: JSON_CREATEDAT ':' JSON_DAY JSON_MONTH JSON_NUM JSON_NUM ':'
JSON_NUM ':' JSON_NUM JSON_NUM ;

user: JSON_USER ':' '{' userbody '}';

id_str: JSON_IDSTR ':' JSON_STRNUM ;

text: JSON_TEXT ':' JSON_STRING ;

other: JSON_STRING ':' JSON_STRING ;

userbody: user_parts | userbody ',' user_parts;

user_parts: id | name | screen_name | location | other;

id: JSON_ID ':' JSON_NUM;

name: JSON_NAME ':' JSON_STRING;

screen_name: JSON_SCREENNAME ':' JSON_STRING;

location: JSON_LOCATION ':' JSON_STRING;

retweeted_status: JSON_RETWEETEDSTATUS ':' '{' rtbody '}';

rtbody: text ',' user | user ',' text;

tweet: JSON_TWEET ':' '{' rtbody '}';

truncated: JSON_TRUNCATED ':' TRUE ',' display_text_range | JSON_TRUNCATED ':' FALSE;

extended_tweet: JSON_EXTENDED_TWEET '{' full_text ',' display_text_range '}' ;

full_text : JSON_FULLTEXT ':' JSON_STRING;

display_text_range: JSON_DISPLAYTEXT RANGE ':' array_num;

```

array_num : '[' nums ']';
nums: JSON_NUM | nums ',' JSON_NUM;
%%

void yyerror(const char *w) {
    printf("*** Error in Line: %d\n",line);
    error++;
}

int main(int argc, char* argv[]) {
    FILE *f;
    if (argc>1)
    {
        f = fopen(argv[1], "r");
        if (!f) {
            printf("Error file %s \n",argv[1]);
            exit(1);
        }
        yyin = f;
    }
    else
        yyin = stdin;

    yyparse();
    printf("\n**\n");
    if (error==0) printf("*** Correct Code\n");
    else printf(" Code Errors ( %d )!\n",error);
}

```

Παράδειγμα:

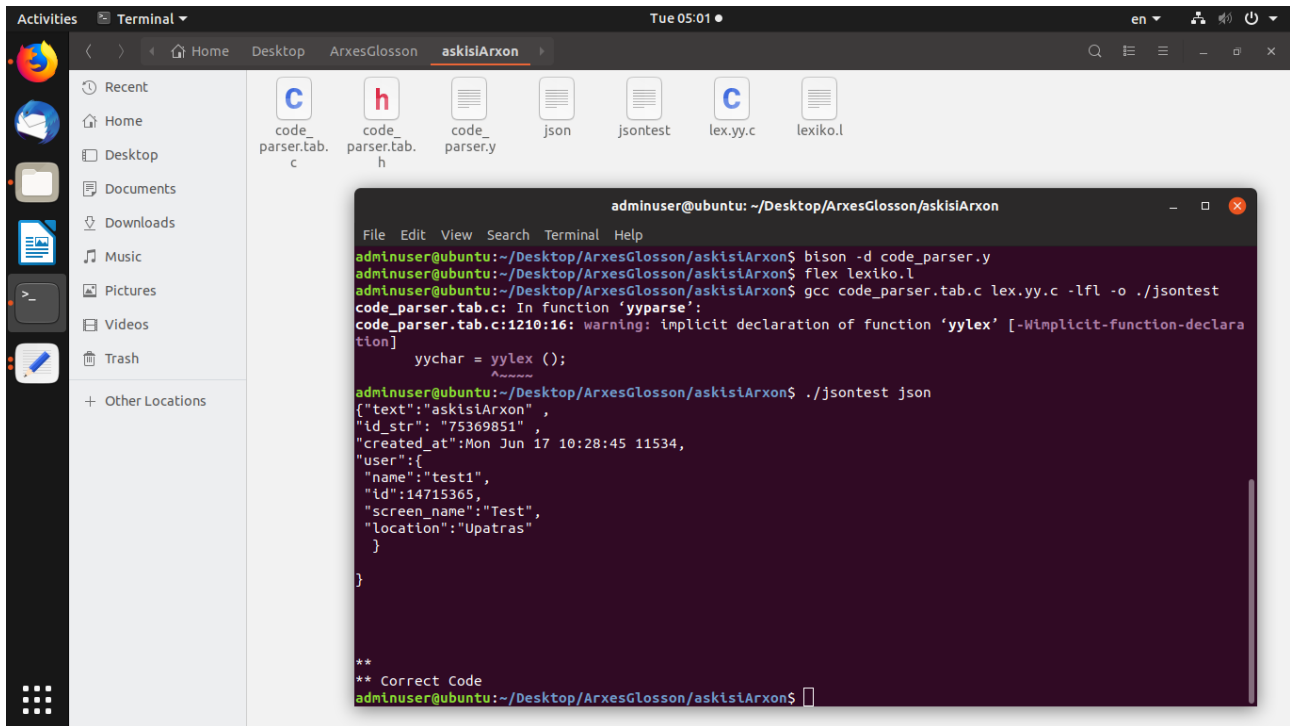
```

{ "text": "askisiArxon" ,
  "id_str": "75369851" ,
  "created_at": "Mon Jun 17 10:28:45 11534",
  "user": {
    "name": "test1",

```

```
"id":14715365,  
"screen_name":"Test",  
"location":"Upatras"  
}  
  
}
```

Αποτέλεσμα:



```
adminuser@ubuntu: ~/Desktop/ArxesGlosson/askisiArxon  
File Edit View Search Terminal Help  
adminuser@ubuntu:~/Desktop/ArxesGlosson/askisiArxon$ bison -d code_parser.y  
adminuser@ubuntu:~/Desktop/ArxesGlosson/askisiArxon$ flex lexiko.l  
adminuser@ubuntu:~/Desktop/ArxesGlosson/askisiArxon$ gcc code_parser.tab.c lex.yy.c -lfl -o ./jsonest  
code_parser.tab.c: In function 'yyparse':  
code_parser.tab.c:1210:16: warning: implicit declaration of function 'yylex' [-Wimplicit-function-declara  
tion]  
    yychar = yylex ();  
                  ^  
adminuser@ubuntu:~/Desktop/ArxesGlosson/askisiArxon$ ./jsonest json  
{  
  "text": "askisiArxon",  
  "id_str": "75369851",  
  "created_at": "Mon Jun 17 10:28:45 11534",  
  "user": {  
    "name": "test1",  
    "id": "14715365",  
    "screen_name": "Test",  
    "location": "Upatras"  
  }  
}  
  
**  
** Correct Code  
adminuser@ubuntu:~/Desktop/ArxesGlosson/askisiArxon$
```