

XX

```
      ###      #####      #####      ###      #####
#####  #  #  #  #  #  #  #  #  #  #  #  #  #
#  #  #  #  #  #  #  #  #  #  #  #  #  #
#####  #  #  #  #  #####  #  #  #####
      #  #  #  #  #  #  #  #  #  #  #  #
#  #  #  #  #  #  #  #  #  #  #  #  #  #
#####  #####  #  #####  #####  #####
```

User: Danilov N.U.

Request id: pr1-320 Printer: pr1

Fri Dec 28 19:34:06 EST 1990

```
program Sintax_Analis;
  Uses Crt,Graph;
  Const max=100;
        maxv=20;
  Type txt=record
        c:array [1..max] of char;
        l:1..max
      end;
  Ptr_to_txt=^txt;
  metasybol=(Alt,Rep,Exi,Con,Ptr,Ide);
  Ptr_to_Gr=^Grammer;
  Grammer=record
        case m:metasybol of
          alt,con:(p1,p2:Ptr to gr);
```



```

begin
    root^.n:=root^.n+1;
    writeln('n:=',root^.n);
    new(pt^.nx);
    pt:=pt^.nx;
end; {while}
pt^.nx:=nil;
Tree_make:=true;
end; {rep}
exi:begin
    Root^.me:=exi;
    new(root^.ep);
    if Tree_make(i,t,g^.p1,root^.p1) and
        Tree_make(i,t,g^.p2,root^.p2)
    then
        root^.e:=true
    else
        begin
            root^.e:=false;
            dispose(root^.ep)
        end; {else}
    Tree_make:=true
end; {exi}
Con:begin
    Root^.me:=Con;
    new(root^.p1);
    new(root^.p2);
    if Tree_make(i,t,g^.p1,root^.p1) and
        Tree_make(i,t,g^.p2,root^.p2)
    then
        Tree_make:=true
    else
        begin
            write(^g,'Concat error in',i)
            end {else}
        end; {Con}
Ptr:begin
    if Tree_make(i,t,g^.p,root^.pp)
    then
        Tree_make:=true
    else
        begin
            Tree_make:=false;
            dispose(root^.pp)
        end {else}
    end; {Ptr}
Ide:begin
    writeln('Identifier');
    writeln('i:=',i);
    writeln('compare: ');
    root^.me:=Ide;
    eq:=true;
    for j:=1 to g^.txtp^.l do
        begin
            writeln(g^.txtp^.c[j], ' with ',t^.c[i+j-1]);
            if (g^.txtp^.c[j]<>t^.c[i+j-1]) or (i+j-1>t^.l)
            then
                eq:=false;
            end; {of for}
        if eq
        then
            begin
                i:=i+g^.txtp^.l;
                Tree_make:=true
            end {then}
        else
            begin
                Tree_make:=false;
                writeln(^g,'Unknown Identifier at ',i)
            end
        end
    end
end

```

```

gramm^p^.txtp^.l=1;
new(text)=
  text^p^.txtp^.l='b';
  text^p^.txtp^.l='b';

```

[illegible]