

Algoritmul 3.1 Descendent_reveniri

```
1: INPUT: G, w =  $x_1x_2\dots x_n$ ;
2: OUTPUT: mesaj (acceptare sau nu), sir_prod
3: s := q; i := 1;  $\alpha := \epsilon$ ;  $\beta := S$ : { configurația inițială}
4: while ( $s \neq t$ ) and ( $s \neq e$ ) do
5:   if  $s = q$  then
6:     if ( $\beta = \epsilon$ ) and ( $i = n+1$ ) then
7:       s:=t
8:     else
9:       if varf( $\beta$ ) = A then
10:        push( $\alpha, A_1$ ); {fie  $A \rightarrow \gamma$  prima producție a lui A }
11:        pop( $\beta, A$ ); push( $\beta, \gamma$ );
12:      else
13:        if varf( $\beta$ ) =  $x_i$  then
14:          i := i+1; push( $\alpha, a$ ); pop( $\beta, a$ )
15:        else
16:          s:=r
17:      else
18:        if s=r then
19:          if varf( $\alpha$ ) = a then
20:            i := i-1; pop( $\alpha, a$ ); push( $\beta, a$ )
21:          else
22:            if  $\exists A \rightarrow \gamma_{j+1}$ , dacă  $A \rightarrow \gamma_j$  a fost ultima folosită then
23:              s:= q;
24:              pop( $\alpha, A_j$ ); push( $\alpha, A_{j+1}$ );
25:              pop( $\beta, \gamma_j$ ); push( $\beta, \gamma_{j+1}$ );
26:            else
27:              if (i=1) and (A=S) then
28:                s:= e
29:              else
30:                pop( $\alpha, A_j$ ); push( $\beta, A$ )
31: if s=e then
32:   mesaj: "EROARE"
33: else
34:   mesaj: "Secvența este acceptată"
35: Construire_sir_prod
```