



ALGORITMI DE REZOLVARE

METODA TRIERII

felicia.burlacu1@gmail.com

Felicia Burlacu , 11D

Cuprins:

1. CE REPREZINTA METODA TRIERII?.....	1
2. PARTICULARITĂȚI DE IMPLEMENTARE:.....	2
3. AVANTAJE SI DEZAVANTAJE:	2
4. PROBLEME REZOLVATE :	3
5. CONCLUZII	7
6. BIBLIOGRAFIE.....	7

1. CE REPREZINTA METODA TRIERII?

Se numește **metoda trierii** o metodă ce indentifică toate soluțiile unei probleme în dependență de mulțimea soluțiilor posibile. Toate soluțiile se identifică prin valori, ce aparțin tipurilor de date studiate: integer, boolean, enumerare, char, subdomeniu, tablouri unidimensionale.

Fie P o problemă, soluția căreia se află printre elementele mulțimii S cu un număr finit de elemente. $S=\{s_1, s_2, s_3, \dots, s_n\}$. Soluția se determină prin analiza fiecărui element si din mulțimea S.

SCHEMA GENERALA :

for i:=1 to k do

if SolutiePosibila (si) then PrelucrareaSolutiei (si)

(SolutiePosibila este o funcție booleana care returneaza valoarea true dacă elementul si satisface condițiile problemei și false în caz contrar, iar PrelucrareaSolutiei este o procedură care efectuează prelucrarea elementului selectat. De obicei, în această procedură soluția si este afișată la ecran.)

Generarea soluțiilor posibile necesită elaborarea unor algoritmi speciali. În general,acești algoritmi realizează operațiile legate de prelucrarea unor mulțimi:

- - reuniunea;

- - intersecția;
- - diferența;
- - generarea tuturor submulțimilor;
- - generarea elementelor unui produs cartezian;
- - generarea permutărilor, aranjamentelor sau combinațiilor de obiecte etc.

2. PARTICULARITĂȚI DE IMPLEMENTARE:

1. Generarea și cercetarea consecutivă a elementelor mulțimii S.
2. Utilizarea funcțiilor și procedurilor pentru fiecare din subproblemele:

Verificarea apartenenței elementului cercetat si la soluție

Plasarea elementului curent în soluție

Generarea următorului element al mulțimii (dacă e necesar)

3. AVANTAJE SI DEZAVANTAJE:

AVANTAJE:

- programele respective sînt relativ simple, iar depanarea lor nu necesită teste sofisticate si la verificare nu trebuie de introdus multe date
- complexitatea temporală a acestor algoritmi este determinată de numărul de elemente k din mulțimea soluțiilor posibile S.
- problemele relativ simple sunt efectuate rapid, incadrindu-se in timpul minim de executie

DEZAVANTAJE

- Întrucît algoritmii exponențiali sunt inacceptabili în cazul datelor de intrare foarte mari, metoda trierii este aplicată numai în scopuri

didactice sau pentru elaborarea unor programe al căror timp de execuție nu este critic.

-Dezavantajul metodei trierii constă

în faptul că timpul cerut de algoritmi respectivi este foarte mare.

4. PROBLEME REZOLVATE :

//Sa se afiseze cate numere patrute perfecte se contin intr-un interval de la 1 la n

```
1. program P1;
2.
3. var
4.   p, n, k: integer;
5.
6. function patrat(p: integer): real;
7. var
8.   x, y: real;
9. begin
10.    x := sqrt(p);
11.    x := round(x); //rotunjeste radacina pana la cel mai
    apropiat nr intreg
12.    y := x * x;
13.    patrat := y;
14. end;
15.
16. function solutieposibila(p: integer): boolean;
17. begin
18.    if patrat(p) = p then solutieposibila := true else
        solutieposibila := false; //verifica daca radacina ridicata la
        patrat este egala cu nr de la 1 la n
19. end;
20.
21. procedure prelucrareasolutiei(p: integer; var k: integer);
22. begin
```

```

23.         if solutieposibila(p) = true then k := k + 1; //daca este
           adevarat atunci numara cate numere sunt patrate perfecte
24.     end;
25.
26.     begin
27.         write('n='); readln(n);
28.         for p := 1 to n do prelucrarea_solutiei(p, k);
29.         writeln(k);
30.     end.
31.
    //CONCLUZIE: prin metoda trierii programul elaborat afiseaza pe
    ecran solutia gasita dupa analiza fiecarui element din intervalul
    dat

```

```

    2//Sa se afiseze numerele dintr-un interval de la 1 la n ale
    caror cifre cand le scazi rezultatul sa fie >0

```

```

1. program p2;
2. var n,i, rsp,x,k:integer;
3.
4. function diferenta(x:integer):integer;
5. var i, d ,er :integer;
6.     s:string;
7. begin
8.     str(x,s);
9.     val(s[1],d,er);
10.
11.     for i:=2 to length(s) do begin
12.         val(s[i],x,er);
13.         d:=d-x;
14.     end;
15.     diferenta:=d;
16. end;
17.
18. function solutieposibila(x:integer):boolean;
19. begin
20.     if diferenta(x) > 0 then solutieposibila:=true;
21. end;
22.
23. procedure prelucraresolutie(x:integer);
24. begin
25.     if solutieposibila(x) = true then writeln(x, ' are
        diferenta cifrelor pozitiva');
26. end;
27.

```

```

28.     begin
29.         write('n='); readln(n);
30.         for x:=1 to n do prelucraresolutie(x);
31.     end.
32.
    //CONCLUZIE: Cu ajutorul metodei trierii putem mai clar si mai
    usor solutiona probleme

3//Sa se afiseze cate numere sunt intr-un interval de la 1 la n
ale caror cifre cand le scazi rezultatul sa fie >0

1. program p3;
2. var n,i,rsp,x,k:integer;
3.
4. function diferenta(x:integer):integer;
5. var i, d ,er :integer;
6.     s:string;
7. begin
8.     str(x,s); //transforma variabila din integer in string
9.     val(s[1],d,er); //transforma prima variabila din string
    inapoi in valoare
10.
11.     for i:=2 to length(s) do begin
12.         val(s[i],x,er);
13.         d:=d-x; //astfel din prima cifra se vor scadea cifrele ce
    il succed
14.     end;
15.     diferenta:=d;
16. end;
17.
18. function solutieposibila(x:integer):boolean;
19. begin
20.     if diferenta(x) > 0 then solutieposibila:=true;
21. end;
22.
23. procedure prelucraresolutie(x:integer; var k:integer);
24. begin
25.     if solutieposibila(x) = true then inc(k);
26. end;
27.
28. begin
29.     write('n='); readln(n);
30.     for x:=1 to n do prelucraresolutie(x,k);
31.     write(k);
32. end.

```

4//Program care vede daca diferenta din cifrele unui numar e
pozitiva sau negativa

```
1. program p4;
2. var i, rsp, x, k: integer;
3.
4. function diferenta(x: integer): integer;
5. var i, d, er: integer;
6.   s: string;
7. begin
8.   str(x, s);
9.   val(s[1], d, er);
10.
11.   for i:=2 to length(s) do begin
12.     val(s[i], x, er);
13.     d:=d-x;
14.   end;
15.   diferenta:=d;
16. end;
17.
18. function solutieposibila(x: integer): boolean;
19. begin
20.   if diferenta(x) > 0 then solutieposibila:=true;
21. end;
22.
23. procedure prelucraresolutie(x: integer);
24. begin
25.   if solutieposibila(x) = true then
26.     writeln('diferenta cifrelor este pozitiva')
27.   else
28.     writeln('diferenta cifrelor este negativa');
29. end;
30.
31. begin
32.   writeln('da numarul'); readln(x);
33.   prelucraresolutie(x);
34. end.
```

5//Sa se verifice daca vectorul are ai mult de doua
elemente pare

```
1. program p3;
2. type vector=array[1..1000] of integer;
3. var a: vector;
4.   i, k, n: integer;
5. function nrelem(a: vector): integer;
6. var i, k: integer;
7. begin
```

```

8.   for i:=1 to n do
9.       if a[i] mod 2 = 0 then inc(k);
10.      nrelem:=k;
11.  end;
12.
13.  function solutieposibila(a:vector):boolean;
14.  begin
15.      if nrelem(a)>2
16.      then solutieposibila:=true;
17.  end;
18.
19.  procedure prelucraresolutie(a:vector);
20.  begin
21.      if solutieposibila(a)=true then writeln('vectorul
are mai mult de 2 elem pare');
22.  end;
23.
24.  begin
25.      writeln('da n'); readln(n);
26.      for i:=1 to n do begin
27.          writeln('introdu elem',i); readln(a[i]);
28.      end;
29.      prelucraresolutie(a);
30.  end.

//CONCLUZIE:o metoda consructiva de a gasi solutii prin
metoda trierii.

```

SURSE: Am avut drept exemplu problemele din manual si conform acestora am inventat probleme proprii.

5. CONCLUZII

Avantajul principal al algoritmilor bazați pe metoda trierii constă în faptul că programele respective sunt relativ simple, iar depanarea lor nu necesită teste sofisticate. Întrucât algoritmii exponențiali sunt inacceptabili în cazul datelor de intrare foarte mari, metoda trierii este aplicată numai în scopuri didactice sau pentru elaborarea unor programe al căror timp de execuție este critic.

6. BIBLIOGRAFIE

1. <https://prezi.com/p/2fundh826js1/metoda-trierii/>
2. <http://blogoinform.blogspot.com/p/metoda-trierii.html>

