Tehnica Greedy

(Realizat de Cuciuc Loredana)

Această metodă presupune că problemele pe care trebuie să le rezolvăm au următoarea

structură:

se dă o mulţime $A=\{a1, a2, ..., an\}$ formată din n elemente;

se cere să determinăm o submulţime B, B se include in A, care îndeplineşte anumite condiţii

pentru a fi acceptată ca soluţie.

Utilizarea.

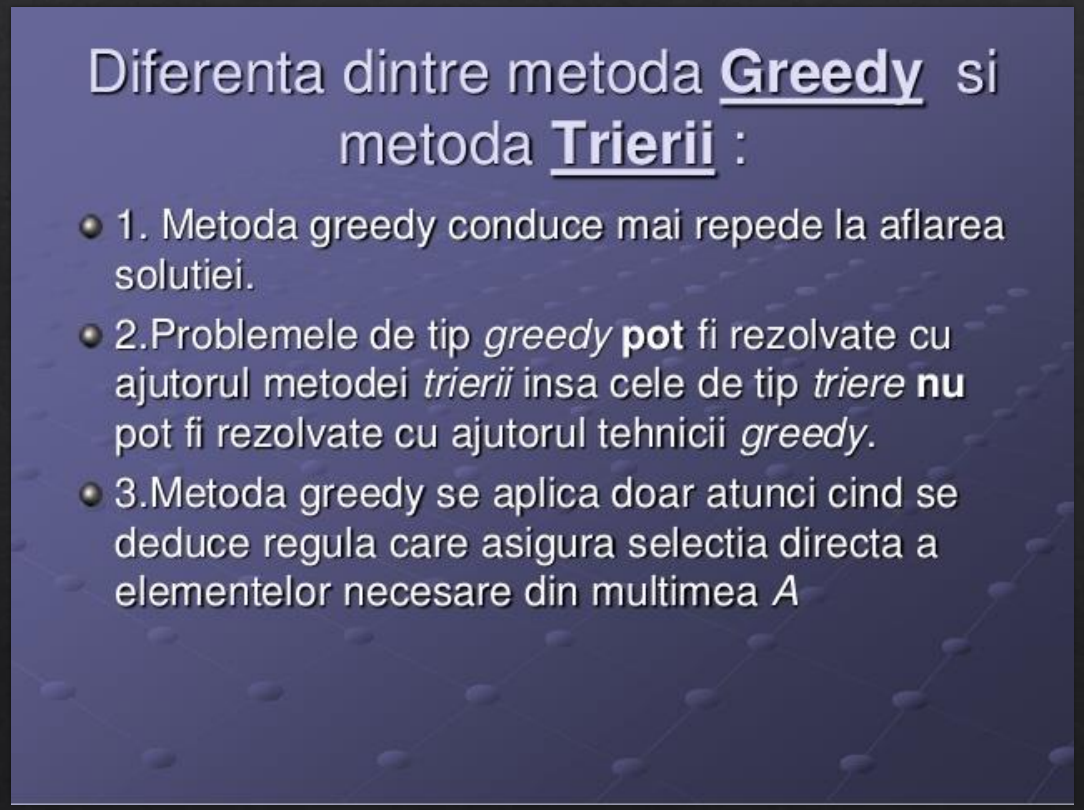
- în metoda *Greedy* se utilizează
- un **criteriu (o regulă)** care asigură alegerea directă a elementelor necesare
- din mulţimea *A*. De obicei, criteriile sau regulile de selecţie nu sînt indicate explicit
- în enunţul problemei şi formularea lor cade în sarcina programatorului. Evident, în
- absenţa unor astfel de criterii metoda *Greedy* nu poate fi aplicată.

Schema generala

- Schema generala a unui algoritm bazat pe metoda *Greedy* poate fi redata cu ajutorul unui ciclu :

```
While ExistaElemente do
Begin
AlegeUnElement(x);
IncludeElementul(x);
End.
```

Avantaje



- După cum se vede, în metoda *Greedy* soluţia problemei se caută prin testarea consecutivă a elementelor din mulţimea *A* şi prin includerea unora din ele în submulţimea *B*. Într-un limbaj plastic, submulţimea *B* încearcă să „înghită” elementele „gustoase”din mulţimea *A*, de unde provine şi denumirea metodei (*greedy* □ lacom, hrăpăreţ).

Elaborati un program care afiseaza elementele mai mari sau egale cu 0 dintr-un sir.

```
Program P1;
var A : array [1..100] of real;
I,n,m : integer
B : array [1..100] of real;
x : real;
Function ExistaElemente : boolean;
var i : integer;
begin
ExistaElemente:=false;
for i:=1 to n do
if A[i]>0then ExistaElemente:=true;
end;
```

```
procedure AlegeUnElement(var x : real);
var i: integer
begin
i:=1;
while A[i]<=0 do i:=i+1;
x:=A[i];
A[i]:=0;
end;
procedure IncludeElementul(x : real);
begin
m:=m+1;
B[m]:=x;
end;
```

Programul principal

```
begin
write('Dați n='); readln(n);
writeln('Dați elementele mulțimii A:');
for i:=1 to n do read(A[i]);
writeln;
m:=0;
while ExistaElemente do
begin
AlegeUnElement(x);
IncludeElementul(x);
end;
writeln('Elementele mulțimii B:');
for i:=1 to m do writeln(B[i]);
readln;
end.
```

Elaborati un program care afiseaza elementele pare dintr-un sir

- **Program** P2;
- **var** A : **array** [1..100] **of** real;
- I,n,m : integer
- B : **array** [1..100] **of** real;
- x : real;
- **Function** ExistaElemente : boolean;
- **var** i : integer;
- **begin**
- ExistaElemente:=false;
- **for** i:=1 **to** n **do**
- **if** A[i]>0**then** ExistaElemente:=true;
- **end**;
- **procedure** AlegeUnElement(**var** x : real);
- **var** i: integer
- **begin**
- i:=1;
- **while** A[i] mod 2=0 **do** i:=i+1;
- x:=A[i];
- A[i]:=0;
- **end**;
- **procedure** IncludeElementul(x : real);
- **begin**
- m:=m+1;
- B[m]:=x;
- **end**;

Programul principal

```
write('Daţi n='); readln(n);
writeln('Daţi elementele mulţimii A:');
for i:=1 to n do read(A[i]);
writeln;
m:=0;
while ExistaElemente do
begin
AlegeUnElement(x);
IncludeElementul(x);
end;
writeln('Elementele mulţimii B:');
for i:=1 to m do writeln(B[i]);
readln;
end.
```

Concluzii
(realizat de Cuciuc Loredana)

Metoda Greedy

Medoda greedy poate folosii în problemele care dîndu-se o mulțime finită A trebuie determinată o mulțime care să îndeplinească anumite condiții.

Metoda data furnizează o singură soluție,reprezentat prin elementul mulțimii S.

Scop- indentificarea problemelor în care solțiile optimă este o submulțime inclusă intr-o submulțime de dată,care trebuie să îndeplinească anumite condiții.