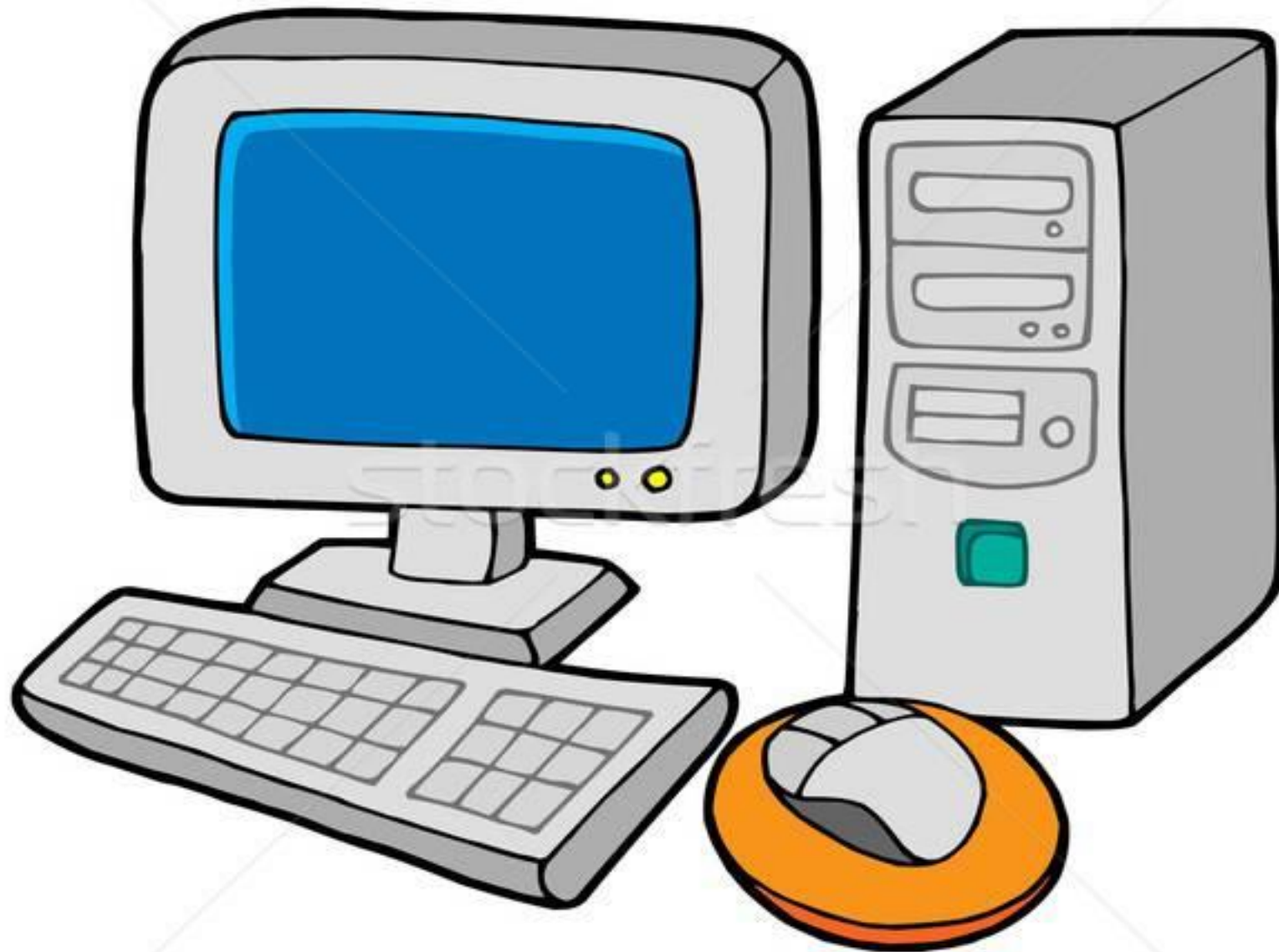
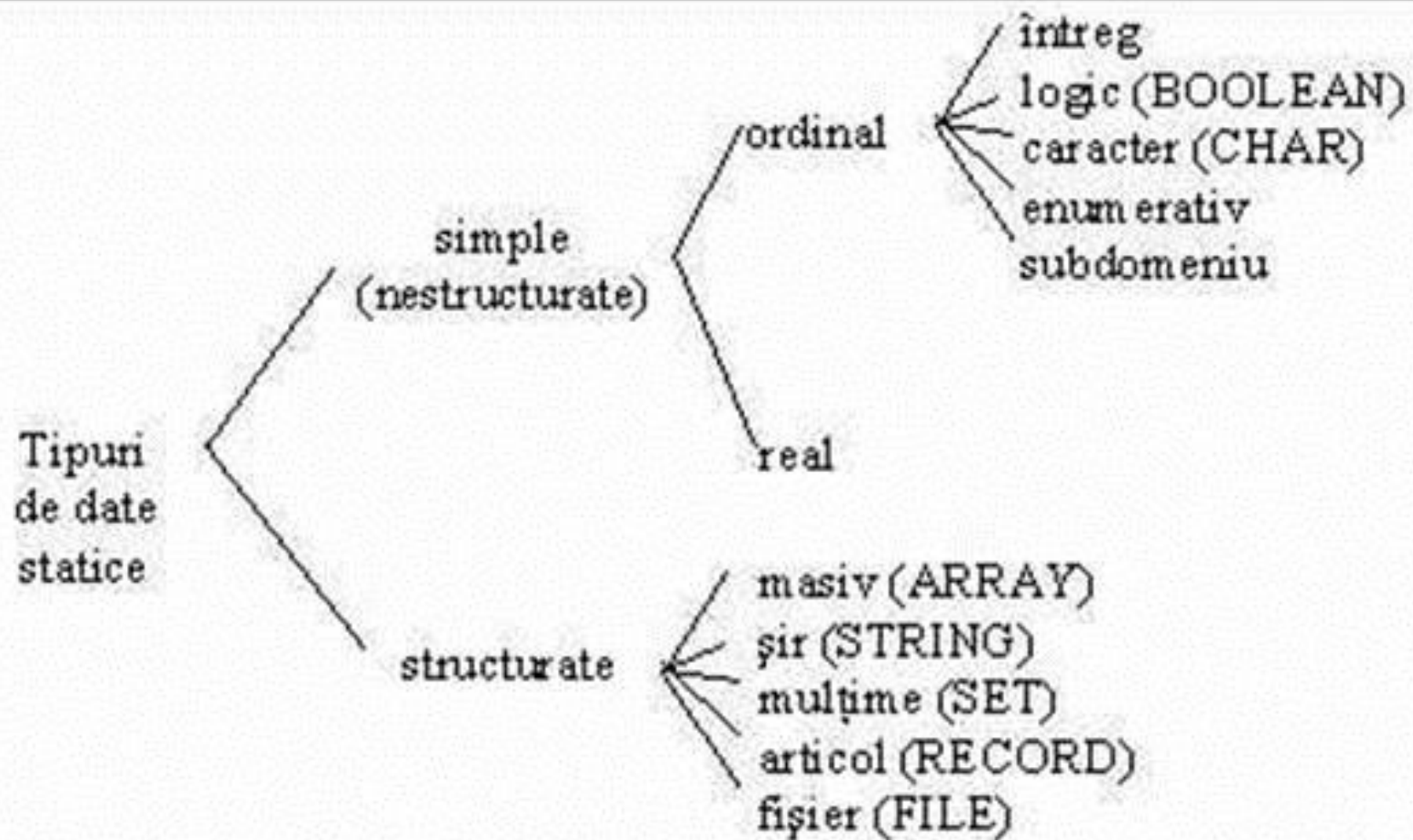


Tipul de date articol(record)

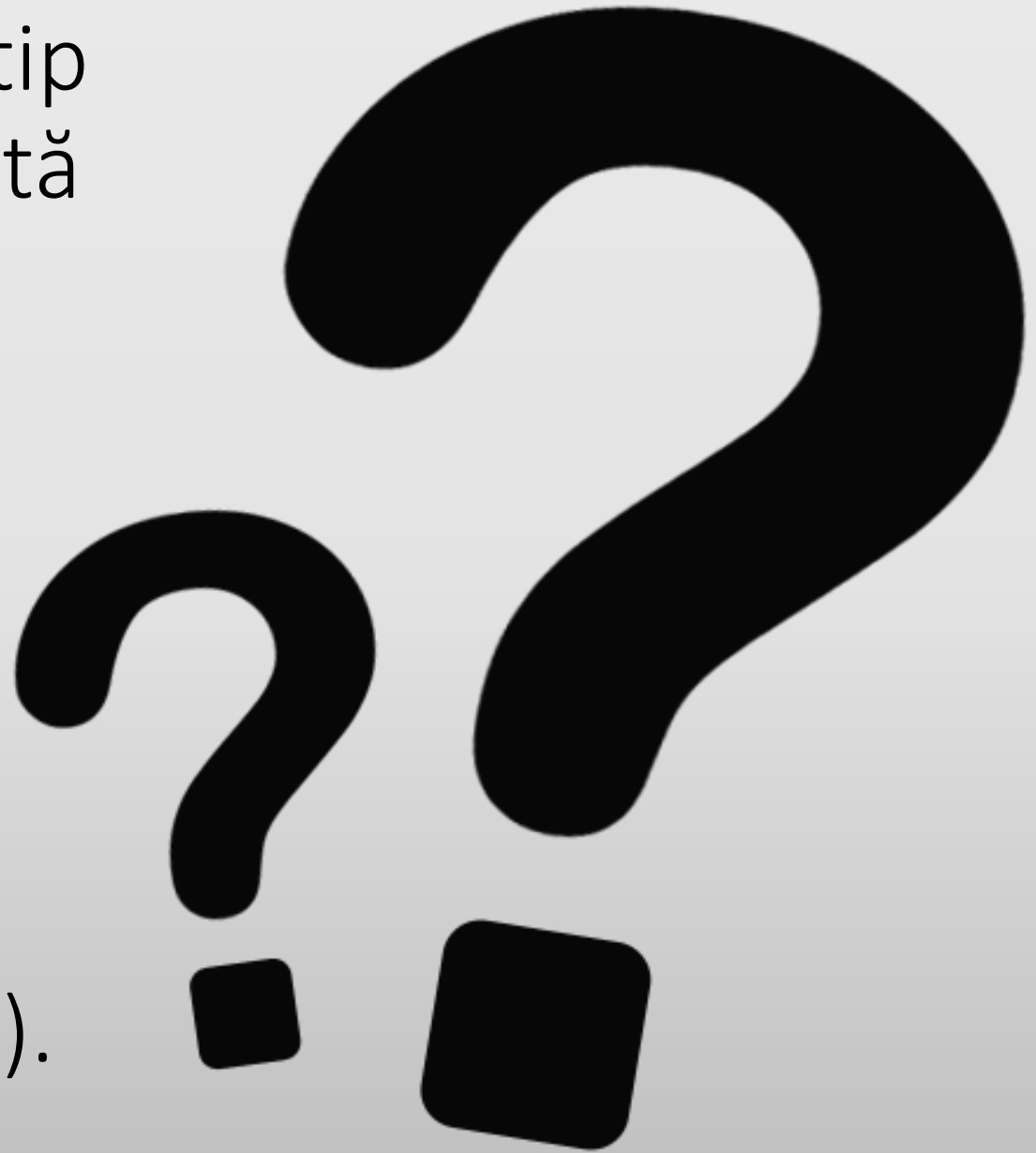


Eleva clasei a Xa "D" – Tataru
Vlada
Institutie Publica Liceul
Teoretic "Spiru Haret"
Profesoara: Maria Guțu



Mulțimea de valori ale unui tip de date **record** este constituită din articol(înregistrări).

Articolele sunt formate din componente, denumite **câmpuri**. Spre deosebire de componentele unui tablou, câmpurile pot fi de tipuri diferite. Fiecare câmp are un nume (identificator de câmp).



Un tip de date articol se definește printr-o structură de forma:

type <nume tip> = record

 <nume câmp 1> : T_1 ;

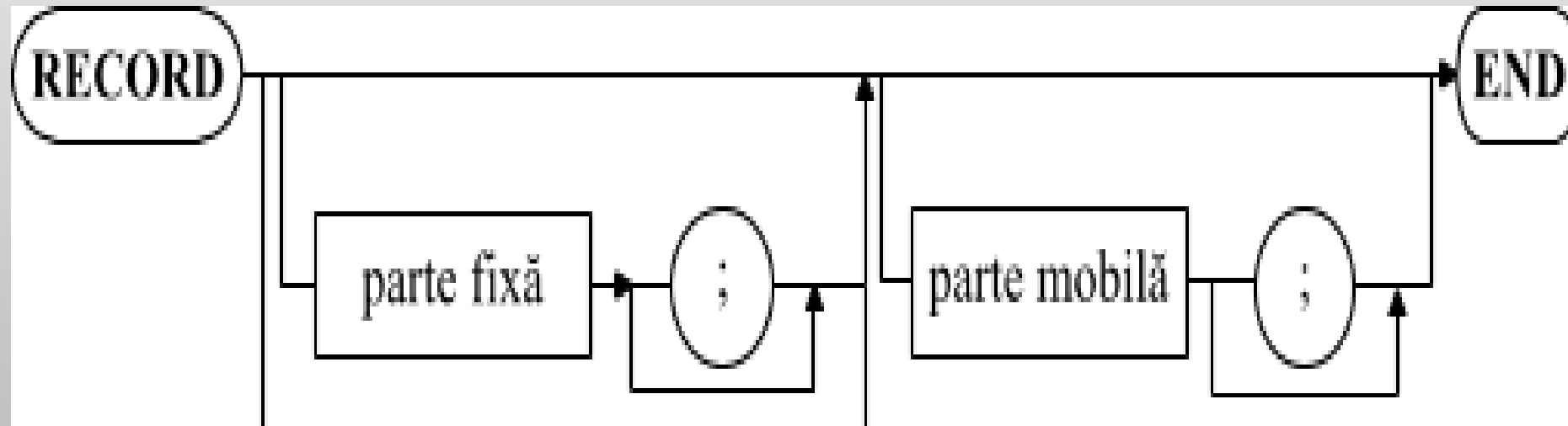
 <nume câmp 2> : T_2 ;

 ...

 <nume câmp n> : T_n ;

end;

unde T_1, T_2, \dots, T_n specifică tipul câmpurilor respective. Tipul unui nume de câmp este arbitrar, astfel un câmp poate să fie la rândul său tot de tip articol. Asupra componentelor datelor de tip articol se pot efectua toate operațiile admise de tipul câmpului respectiv. Orice tip de date articol poate servi ca tip de bază pentru formarea altor tipuri structurate.



Fiind date două variabile de tip *articol* de același tip, numele variabilelor pot apărea într-o instrucțiune de atribuire. Această atribuire înseamnă copierea tuturor câmpurilor din membrul drept în membrul stâng. De exemplu, pentru tipul de date și variabilele declarate mai sus instrucțiunile

```
E1:=E2;  
T2:=T3;  
P2:=P1
```

, sunt corecte.

Fiecare componenta a unei variabile de tip **record** poate fi specificată explicit, prin numele variabilei și denumirile de câmpuri, separate prin puncte.

Exemple:

```
1)E1.Nume, E1.Prenume, E1.NotaMedie;  
2)E2.Nume, E2.Prenume, E2.NotaMedie;
```

Record este un tip special de structură a datelor care, spre deosebire de array, colectează o serie de domenii de tipuri diferite de date care definesc o anumită structură, cum ar fi o carte, un produs, o persoană și multe altele. Programatorul definește structura de date sub definiția de type.

```
Program P3;  
type data=record  
    z:1..31;  
    l:1..12;  
    a:integer;  
end;  
persoana=record  
    np:string;  
    dn:data;  
end;  
lista=array[1..50] of persoana;  
var n,i,z:integer;  
    lp:lista;
```

```
begin  
    Write('Introdu numarul  
    persoanelor:'); readln(n);  
    for i:=1 to n do with lp[i] do  
        begin writeln;  
            writeln('Introdu datele  
            persoanei:',i);  
            write('Nume Prenume: ');  
            readln(np);  
            write('data nasterii: ');  
            readln(dn.z);  
            write('luna nasterii: '); readln(dn.l);  
            write('anul nasterii: ');  
            readln(dn.a); end;  
    end;
```

```
begin  
    write('Introdu numar  
    linei pe care doresti sa o  
    vizualizezi: ');  
    readln(i);  
    writeln(lp[i].np,',',lp[i].d  
    n.z, '/', lp[i].dn.l, '/', lp[i].d  
    n.a);  
end;  
readln;  
end
```

Concluzie

Utilizarea tipului de date structurat articol(record) permite lucrul cu o cantitate mai mare de date în comparație cu alte tipuri de date structurate studiate până acum. Un avantaj mare pentru tipul articol reprezintă faptul că el poate fi utilizat aproximativ ca un tablou, dar câmpurile sale, spre deosebire de elementele tabloului, pot fi de tipuri total diferite.