

```
program Fisiere_cu_tip;

uses Crt;

{ Definim mai multe tipuri de date si vom accesa unul si acelasi
  fisier pe disc pentru a citi fiecare din aceste tipuri de date. }
type
  TReal = record
      x : Real;
  end;

  TNatural = record
      n : Integer;
  end;

var
  { Pentru fiecare din tipurile de date definite mai sus creem cate
    o variabila de tip fisier de tipul respectiv. Totodata mai definim
    cate o variabila de lucru de tipul respectiv. }
  r : TReal;
  fr : file of TReal;

  n : TNatural;
  fn : file of TNatural;

  { Mai definim o variabila de tip fisier de tip char. }
  c : Char;
  fc : file of Char;

begin
  ClrScr;

  { Intai vrem sa vedem cat spatiu ocupa in memorie tipurile de baza. }
  WriteLn('Dimensiune Integer: ', SizeOf(Integer));
  WriteLn('Dimensiune Real: ', SizeOf(Real));
  WriteLn;

  { Apoi vrem sa vedem cat ocupa in memorie tipurile definite de noi. }
  WriteLn('Dimensiune TReal: ', SizeOf(TReal));
  WriteLn('Dimensiune TNatural: ', SizeOf(TNatural));
  WriteLn;

  { Cream un fisier de tip TReal si scriem in el doua inregistrari. }
  Assign(fr, 'fis.dat');
  Rewrite(fr);
  r.x := 1.1;
  Write(fr, r);
  r.x := 2.256;
  Write(fr, r);
  Close(fr);

  { Redeschidem fisierul de tip TReal. Aplicam operatorul FileSize
    pentru a vedea cate inregistrari sunt in el. Apoi citim toate
    inregistrările pana intalnim EOF si afisam ce am citit. }
  Reset(fr);
  WriteLn('Numar inregistrari in fisier TReal: ', FileSize(fr));
  while not(eof(fr)) do
      begin
          Read(fr, r);
          WriteLn(r.x:0:2);
      end;
end;
```

```
Close(fr);
WriteLn;

{ Acum deschidem acelasi fisier de pe disc, dar ca si fisier de tip
  Char. Aplicam din nou operatorul FileSize pentru a vedea cate
  inregistrari sunt in fisier. Cum dimensiunea unei inregistrari Char
  este de 1 octet, vom obtine practic dimensiunea pe disc a fisierului.
  Apoi citi rand pe rand caractere din fisier si le afisam pe ecran. }
Assign(fc, 'fis.dat');
Reset(fc);
WriteLn('Numar inregistrari in fisier Char: ', FileSize(fc));
while not(eof(fc)) do
begin
    Read(fc, c);
    { Afisam caracterul ca si intreg, folosind functia "Ord".
      Este foarte probabil ca in fisier sa intalnim caractere
      netiparibile, pe care nu le-am putea afisa ca si
      caractere normale. }
    Write(ord(c), ' ');
end;
Close(fc);
WriteLn;
WriteLn;

{ Mai deschidem o data tot acelasi fisier, dar acum ca fisier de
  tipul TNatural. Din nou aplicam operatorul FileSize pentru a vedea
  cate inregistrari se gasesc in fisier. Apoi citim toate inregistrarile
  rand pe rand si le afisam, pana la intalnirea EOF. }
Assign(fn, 'fis.dat');
Reset(fn);
WriteLn('Numar inregistrari in fisier TNatural: ', FileSize(fn));
while not(eof(fn)) do
begin
    Read(fn, n);
    WriteLn(n.n);
end;
Close(fn);

{ Ideea este ca un fisier reprezinta o structura foarte flexibila, pe
  care o putem accesa in ce mod vrem noi. Doar ca nu in orice mod de
  accesare informatiile obtinute din fisier au sens. De aceea este
  important ca sa citim informatii din fisiere exact in acelasi
  format in care ele au fost scrise. }
end.
```