

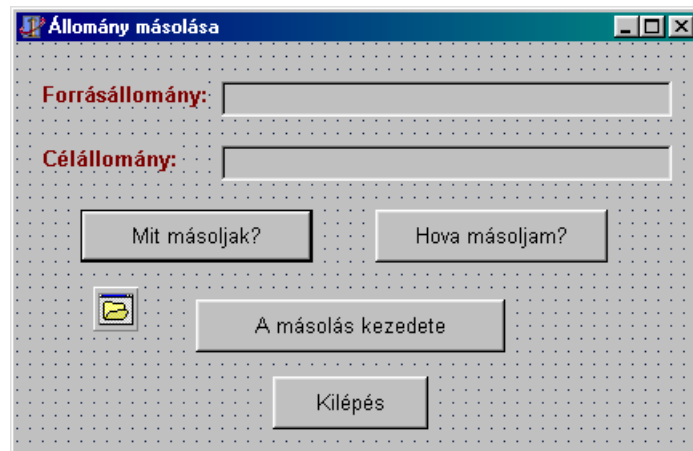
7.2. Adatfolyamok (streams) kezelése

1. Állományok másolása adatfolyammal [FMasolas](#)
2. Állomány- és adatfolyam kezelés összehasonlítása [FajlAtlag](#)



Készítsünk állománymásoló programot. A másolás végrehajtásához használjunk adatfolyamokat!
(FMasolas)

Tervezzük meg az alkalmazás felhasználói felületét!



A „Mit másoljak?” gomb megnyomásakor megjelenő általános fájlnyitás párbeszédablakkal kiválasztjuk azt az állományt, amit másolni szeretnénk - az állomány neve az *Edit1* vezérlőbe is bekerül.

```
procedure TForm1.btnMitClick(Sender: TObject);
begin
    // Szűrő beállítása *.*
    OpenFileDialog1.Filter := 'Minden fájl (*.*)|*.*|';
    OpenFileDialog1.FilterIndex := 1;
    OpenFileDialog1.Title := 'A forrásállomány kiválasztása';
    // Az állomány nevének kiválasztása után a fájl nevével
    // tárolása a formon
    Edit1.Text := '';
    if OpenFileDialog1.Execute then
        Edit1.Text := OpenFileDialog1.FileName;
end;
```

A „Hova másoljam?” gomb megnyomásakor a másolás célját választhatjuk ki, ami az *Edit2* vezérlőbe is belekerül.

```
procedure TForm1.btnHovaClick(Sender: TObject);
begin
    // Szűrő beállítása *.*
    OpenFileDialog1.Filter := 'Minden fájl (*.*)|*.*|';
    OpenFileDialog1.FilterIndex := 1;
    OpenFileDialog1.Title := 'A célállomány kiválasztása';
    // Az állomány nevének kiválasztása után a fájl nevével
    // tárolása a formon
    Edit2.Text := '';
    if OpenFileDialog1.Execute then
        Edit2.Text := OpenFileDialog1.FileName;
end;
```

A másolás a „*Másolás kezdete*” gombbal indítható. Gondoskodunk arról is, hogy a DOS rendszerben megszokott módon célnak elegendő legyen csak egy könyvtárat választani, ilyenkor a másolt állomány neve nem változik. Az esetleges hibákat kivételeken keresztül kezeljük.

```
procedure TForm1.btnMasolasClick(Sender: TObject);
var
    Fnev : string;
    i     : integer;
begin
    if (Edit1.Text<>'' ) and (Edit2.Text<>'' ) then
        begin
            // Ha a cél útvonal, az állománynév hozzáillesztése
            for i:=length(Edit1.Text) downto 1 do
                if Edit1.Text[i]='\ ' then Break;
            if Edit1.Text[i]='\ ' then
                Fnev:=copy(Edit1.Text, i+1, length(Edit1.Text)-i);
            if Edit2.Text[length(Edit2.Text)]='\ ' then
                Edit2.Text:=Edit2.Text+Fnev;
            try
                CopyIt(Edit1.Text, Edit2.Text);
            except
                MessageDlg('Hiba lépett fel a másolás során!', mtError, [mbOK],0);
                Edit1.Setfocus;
                Exit;
            end;
            MessageDlg('A másolás megtörtént!', mtInformation, [mbOK],0);
            Edit1.Text:='';
            Edit2.Text:='';
            btnMit.Setfocus;
        end
    else
        begin
            MessageBeep(0);
            MessageDlg('Rossz, vagy hiányzó paraméterek!', mtError, [mbOK],0);
        end;
end;
```

A másolás a *CopyIt* eljárással történik, adatfolyamok segítségével.

```
procedure CopyIt( const FromName, ToName: string );
var
    ff, tf: TFileStream;
begin
    ff := TFileStream.Create( FromName, fmOpenRead );
    try
        tf := TFileStream.Create( ToName, fmOpenWrite or fmCreate );
        try
            tf.CopyFrom(ff, ff.Size);
        finally
            tf.Free;
        end;
    finally
        ff.Free;
    end;
end;
```



Hasonlítsuk össze adott mennyiségű lemezen tárolt adat feldolgozásához szükséges időket, ha az adatokat állományként, lemez-, illetve memóriafolyamként kezeljük! (*FajlAtlas*)

Az összehasonlítást valós adatok átlagának kiszámításával, a hagyományos, a *TFileStream*- és a *TMemoryStream* alapú fájlkezelés esetére végezzük el.

A fájlba írt adatok mennyiségét az *edt* szövegszerkesztővel, illetve csatlakoztatott *UpDown1* vezérlővel adjuk meg. A feldolgozás idejét mérő gombok a fájl elmentéséig nem aktívak, erről a form létrehozásakor is gondoskodunk.

```
procedure TForm1.edtAdatmennyisegChange(Sender: TObject);
begin
  btnFileOf.enabled:=false;
  btnFStream.enabled:=false;
  btnMStream.enabled:=false;
  edtLetrehoz.clear;
  edtFileof.clear;
  edtFStream.clear;
  edtMStream.clear;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
  btnFileOf.enabled:=false;
  btnFStream.enabled:=false;
  btnMStream.enabled:=false;
  UpDown1.thousands:=false;
end;
```

A „*Létrehoz Fájlként*” gomb megnyomásakor a *forgalom.dat* állományba a megadott mennyiségnek (*db*) megfelelő mennyiségű véletlen valós számot írunk. A hibás adatmegadást kivételkezeléssel szűrjük ki.

```
procedure TForm1.btmLetrehozClick(Sender: TObject);
var f:file of real;
    i:integer;
    adat:real;
    db:integer;
begin
  Randomize;
  edtLetrehoz.clear;
  edtFileof.clear;
  edtFStream.clear;
  edtMStream.clear;
  caption:='Start'; Beep;
  idomero(start, time);
  assignfile(f, 'forgalom.dat');
  rewrite(f);
```

```

try
  db:=1024*strtoint(edtAdatmennyiseg.text) div sizeof(real);
except
  Showmessage('Hibás adatmegadás');
  exit;
end;
if db<0 then
begin
  Showmessage('Hibás adatmegadás');
  exit;
end;
for i:=1 to db do
begin
  adat:=(100000+random(maxint))/random(maxint);
  write(f,adat);
end;
closefile(f);
edtLetrehoz.text:=floattostr(idomero(stop, time))+ ' mp';
caption:='Stop'; Beep;
btnFileOf.enabled:=true;
btnFStream.enabled:=true;
btnMStream.enabled:=true;
end;

```

Fájlként történő feldolgozáshoz a *btnFileof* gombot használjuk. A fájl végét figyelve a beolvasott adatokat összeadjuk, majd átlagoljuk. Az időmérést az *idomero* függvény segítségével végezzük.

```

procedure TForm1.btnFileofClick(Sender: TObject);
var f:file of real;
    i:integer;
    sum, adat : real;
begin
  caption:='Start'; Beep;
  idomero(start, time);
  assignfile(f, 'forgalom.dat');
  filemode:=0;
  reset(f);
  i:=0;
  sum:=0;
  while not eof(f) do
    begin
      read(f,adat);
      inc (i);
      sum:=sum+adat;
    end;
  closefile(f);
  caption:='Stop'; Beep;
  edtFileOf.text:=floattostr(idomero(stop, time))+ ' mp';
  caption:=floattostr(sum/i);
end;

```

A lemezes adatfolyam esetén a *btnFStream* gombot használjuk, melynek eseménykezelőjében az előzőhöz hasonló módon járunk el. Az *fs* adatfolyam méretét használjuk az olvasás vezérlésére.

```

procedure TForm1.btnFStreamClick(Sender: TObject);
var
  fs : TFileStream;
  i,db : integer;
  sum, adat : real;
begin
  caption:='Start'; Beep;
  idomero(start, time);
  fs:=TFileStream.create('Forgalom.dat', fmOpenread);
  sum:=0;;
  db:=fs.Size div sizeof(real);
  for i:=1 to db do
    begin
      fs.read(adat,sizeof(real));
      sum:=sum+adat;
    end;
  end;

```

```

fs.Free;
caption:='Stop'; Beep;
edtFStream.text:=floattostr(idomero(stop, time))+ ' mp';
caption:=floattostr(sum/db);
end;

```

A memória adatfolyammal történő olvasást a *btnMStream* gomb eseménykezelője végezzük. Az *fs* fájl adatfolyamból az *ms* memória adatfolyamba a **CopyFrom** metódussal pumpáljuk át az adatokat és az összegzést pointerrel végezzük.

```

procedure TForm1.btnMStreamClick(Sender: TObject);
var
  ms : TMemoryStream;
  fs : TFileStream;
  p : ^ real;
  i,db : integer;
  sum : real;
begin
  caption:='Start'; Beep;
  idomero(start, time);
  ms:=TMemoryStream.create;
  fs:=TFileStream.create('Forgalom.dat', fmOpenread);
  ms.SetSize(fs.Size);
  ms.CopyFrom(fs,fs.Size);
  p:=ms.Memory;
  sum:=0;;
  db:=fs.Size div sizeof(real);
  for i:=1 to db do
    begin
      sum:=sum+p^;
      inc(p);
    end;
  fs.Free;
  ms.free;
  caption:='Stop'; Beep;
  edtMStream.text:=floattostr(idomero(stop, time))+ ' mp';
  caption:=floattostr(sum/db);
end;

```

Az időmérő vezérléséhez felsorolt típust definiálunk

```

type TSS=(start, stop);

```

Az időmérő függvény úgy működik, mint egy stopper, el lehet indítani, meg lehet állítani és megállítás esetén az eltelt időt szolgáltatja másodpercben.

```

function idomero(ss: TSS; ido: tdatetime):real;
const
  starttime:Tdatetime=0;
var
  diff:integer;
  s1, s2 : TTimeStamp;
begin
  result:=0;
  case ss of
    start: starttime:=ido;
    stop:
      begin
        s1:=dateTimeToTimeStamp(starttime);
        s2:=dateTimeToTimeStamp(ido);
        diff:=s2.time-s1.time;
        result:=diff/1000;
      end;
  end;
end;
end;

```

A program futási képe jól szemlélteti a szükséges időráfordítások arányát.

