

Teoriako ariketa

```
package datuak is
  type Ikasle is record
    Izena: String(1..10);
    Abizena: String(1..10);
  end record;
  type Ikasle_Bektorea is array (1..120) of Ikasle;
  type Ikasle_Zerrenda is record
    Zenbat: Integer;
    Ikasleak: Ikasle_Bektorea;
  end record;

end datuak;
```



Teoriako ariketa

```
with datuak, ada.text_io;  
use datuak, ada.text_io;  
with egitura_bete, ikaslea_ezabatu;  
  
procedure ikasleak_kudeatu is  
  Klaseko_ikasleak: Ikasle_zerrenda;  
  Izena: String(1..10);  
begin  
  Klaseko_ikasleak.Zenbat:= 0;  
  egitura_bete(Klaseko_ikasleak);  
  put_line("ezabatzeko ikaslearen izena:(" return amaitzeko)");  
  get(Izena);  
  loop exit when Izena = "          ";  
    ikaslea_ezabatu(Klaseko_ikasleak, Izena);  
    put_line("ezabatzeko ikaslearen izena:(" return amaitzeko)");  
    get(Izena);  
  end loop;  
end ikasleak_kudeatu;
```

Teoriako ariketa

```
with datuak, ada.text_io;
use datuak, ada.text_io;
procedure egitura_bete(IZ: out Ikasle_Zerrenda) is
    Ize, Abi: String(1..10);
begin
    put_line("sartu izena/return amaitzeko");
    get(Ize);
    put_line("sartu abizena");
    get(Abi);
    loop exit when Ize = "          " ; - - 10 hutsune
        IZ.Zenbat := IZ.Zenbat + 1;
        IZ.Ikasleak(IZ.Zenbat).Izena:= Ize;
        IZ.Ikasleak(IZ.Zenbat).Abizena:= Abi;
        put_line("sartu izena/return amaitzeko");
        get(Ize);
        put_line("sartu abizena");
        get(Abi);
    end loop;
    put_line("egitura beteta");
end egitura_bete;
```



Teoriako ariketa

```
with datuak;
```

```
use datuak;
```

```
with posizioa, ezabatu;
```

```
procedure ikaslea_ezabatu(Eguneratze_ikasleak: in out Ikasle_Zerrenda;  
Ezabatzeko_Izena: in String) is
```

```
    Ind: Integer;
```

```
begin
```

```
    Ind := posizioa(Eguneratze_ikasleak, Ezabatzeko_izena);
```

```
    ezabatu(Eguneratze_ikasleak, Ind);
```

```
end ikaslea_ezabatu;
```

Teoriako ariketa

```
with datuak;  
use datuak;  
  
function posizioa (IZ: in Ikasle_Zerrenda; Izena: in String) return Integer is  
    Indizea: Integer;  
    Aurkitua: Boolean := False;  
begin  
    Indizea:=1;  
    loop exit when Indizea > IZ.Zenbat or Aurkitua;  
        if IZ.Ikasleak(Indizea).Izena = Izena then  
            Aurkitua:= True;  
        else  
            Indizea:= Indizea+1;  
        end if;  
    end loop;  
    return Indizea;  
end posizioa;
```

Teoriako ariketa

```
with datuak;
```

```
use datuak;
```

```
with ezkerrera_mugitu;
```

```
procedure ezabatu(IZ: in out Ikasle_zerrenda; Pos: in Integer) is
```

```
begin
```

```
    ezkerrera_mugitu(IZ, Pos);
```

```
    IZ.Zenbat:= IZ.Zenbat - 1;
```

```
end ezabatu;
```



Teoriako ariketa

```
with datuak;
```

```
use datuak;
```

```
procedure ezkerrera_mugitu(IZ: in out Ikasle_Zerrenda; Ind: in Integer) is
```

```
    Indizea: Integer:= Ind;
```

```
begin
```

```
    loop exit when Indizea = IZ.Zenbat;
```

```
        IZ.Ikasleak(Indizea):= IZ.Ikasleak(Indizea+1);
```

```
        Indizea:= Indizea + 1;
```

```
    end loop;
```

```
end ezkerrera_mugitu;
```



Proposatutako ariketa 1

```
package datuak is
  type km_bektorea is array(1..5) of integer;

  type saltzaile is record
    identifikazioa: integer;
    izena: string(1..6);
    abizena:string(1..6);
    kmetroak: km_bektorea;
  end record;

  type saltzaileen_bektorea is array(1..10) of saltzaile;

  type T_emaiza is record --emaiza itzultzeko
    izena: String(1..6);
    km_max: integer;
  end record;
end datuak;
```


Proposatutako ariketa 1

```
with ada.text_io, ada.integer_text_io, datuak;  
use ada.text_io, ada.integer_text_io, datuak;
```

```
procedure lista_bete (l: out saltzaileen_bektorea) is
```

```
    id: integer;  
    izena, abizena: string(1..6);  
    kilom: integer;
```

```
begin
```

```
    put("lista betetzera goaz:");
```

```
    new_line;
```

```
    for i in 1..l'last loop
```

```
        put ("Sartu identifikadorea saltzaile("&integer'image(i)&"): ");
```

```
        new_line;
```

```
        get(id);
```

```
        put ("Sartu izena saltzaile("&integer'image(i)&"): (6 KARAKTERE SARTU BEHAR DITUGU)");
```

```
        new_line;
```

```
        get(izena);
```

```
        put ("introduzca el abizena del saltzaile("&integer'image(i)&"): (6 KARAKTERE SARTU BEHAR DITUGU)");
```

```
        new_line;
```

```
        get(abizena);
```

```
        l(i).identifikazioa:=id;
```

```
        l(i).izena:=izena;
```

```
        l(i).abizena:=abizena;
```

```
        for j in 1..l(i).kmetroak'last loop
```

```
            put("sartu km( "&integer'image(j)&"):");
```

```
            new_line;
```

```
            get(kilom);
```

```
            l(i).kmetroak(j):=kilom;
```

```
        end loop;
```

```
    end loop;
```

```
    new_line;
```

```
end lista_bete;
```

eman ta zabal zazu



Proposatutako ariketa 1

with datuak;

use datuak;

function **km_handi_eta_izena** (l: in saltzaileen_bektorea) return T_emaiza is

ema:T_emaiza;

begin

ema.km_max:=0;

for i in 1..l'last loop

for j in 1..l(i).kmetroak'last loop

if (ema.km_max<l(i).kmetroak(j)) then

ema.km_max:=l(i).kmetroak(j);

ema.izena:=l(i).izena;

end if;

end loop;

end loop;

return ema;

end **km_handi_eta_izena**;

Proposatutako ariketa 1

```
with datuak, ada.text_io; use ada.text_io, lista_bete, km_handi_eta_izena;
use datuak, ada.text_io; use ada.text_io;
```

procedure nagusia is

```
  lista: saltzaileen_bektorea;
```

```
  -- lista: datuak.saltzaileen_bektorea; -- "USE DATUAK" IPINI EZ DUGUNEZ, HELBIDEA IPINI BEHAR DUGU
```

```
  ema: T_emaiza;
```

begin

```
  --lista_bete(lista);
```

```
  --DATUAK MANUALKI EDO AUTOMATIKOKI SARTU NAHI DUZUEN AUKERA DEZAKEZUE.
```

```
  lista(1).identifikazioa:=1; lista(2).identifikazioa:=2; lista(3).identifikazioa:=3; lista(4).identifikazioa:=4; lista(5).identifikazioa:=5;
```

```
  lista(6).identifikazioa:=6; lista(7).identifikazioa:=7; lista(8).identifikazioa:=8; lista(9).identifikazioa:=9; lista(10).identifikazioa:=10;
```

```
  lista(1).izena:="pepe1 "; lista(2).izena:="pepe2 "; lista(3).izena:="pepe3 "; lista(4).izena:="pepe4 "; lista(5).izena:="pepe5 ";
```

```
  lista(6).izena:="pepe6 "; lista(7).izena:="pepe7 "; lista(8).izena:="pepe8 "; lista(9).izena:="pepe9 "; lista(10).izena:="pepe10";
```

```
  lista(1).abizena:="garcia"; lista(2).abizena:="garcia"; lista(3).abizena:="garcia"; lista(4).abizena:="garcia"; lista(5).abizena:="garcia";
```

```
  lista(6).abizena:="garcia"; lista(7).abizena:="garcia"; lista(8).abizena:="garcia"; lista(9).abizena:="garcia"; lista(10).abizena:="garcia";
```

```
  lista(1).kmetroak:= (1111,2,3,4,5);   lista(2).kmetroak:= (11,12,13,14,15);   lista(3).kmetroak:= (21,22,23,24,25);   lista(4).kmetroak:= (31,32,33,34,35);
```

```
  lista(5).kmetroak:= (41,42,143,44,45);
```

```
  lista(6).kmetroak:= (51,52,53,54,55);   lista(7).kmetroak:= (61,62,63,64,65);   lista(8).kmetroak:= (71,72,73,74,75);   lista(9).kmetroak:=
(81,82,83,84,85);
```

```
  lista(10).kmetroak:= (91,92,93,94,95);
```

```
  ema:=km_handi_eta_izena(lista);
```

```
  put("1) Saltzailearen izena "& ema.izena &" da eta "&integer'image(ema.km_max)&" kilometro ditu");
```

end nagusia;

eman ta zabal zazu

