

FLUXU-DIAGRAMAK: oinarrizko egiturak

1) IF

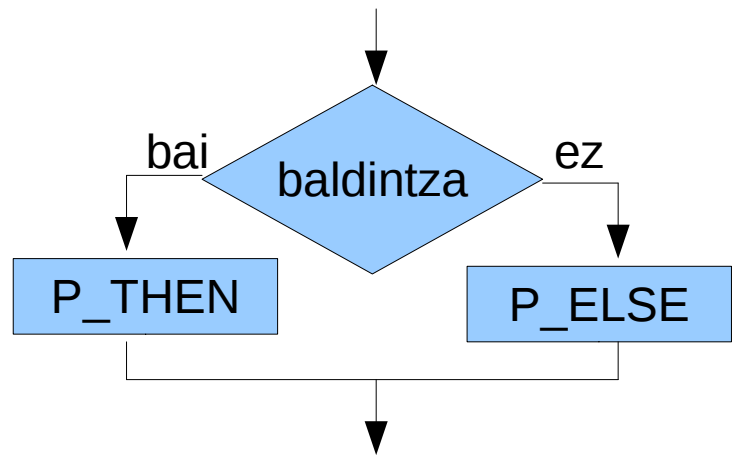
if baldintza then

P_THEN

else

P_ELSE

next

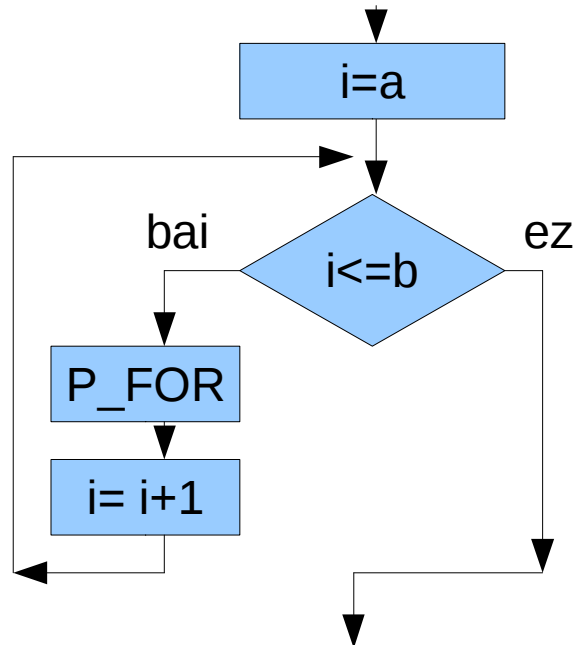


2) FOR

for i=a to b

P_FOR

next

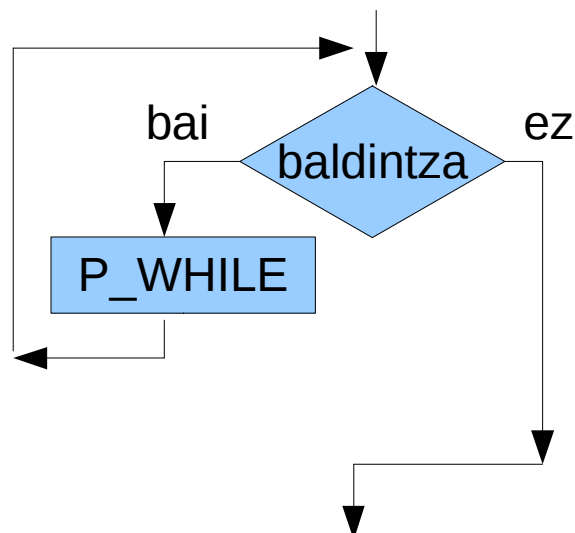


3) WHILE

while baldintza

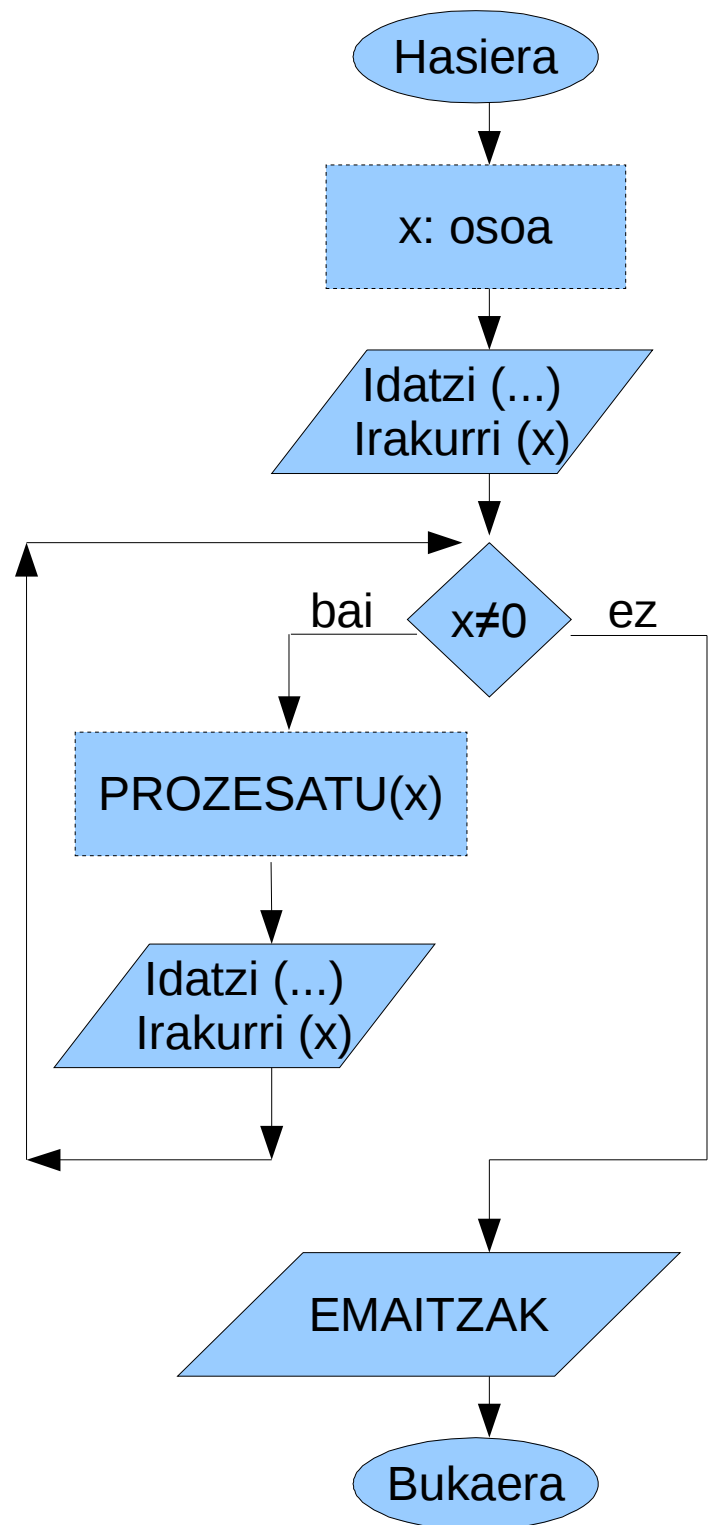
P_WHILE

wend



1) Zeroz bukatutako zenbaki sekuentzia bateko maximoa kalkulatu:

```
Dim x as Integer  
x= InputBox(...)  
while x<>0  
    PROZESATU (x)  
    x= InputBox(...)  
wend  
EMAITZAK
```



2) Bektore bat:

Dim Bektorea as Integer(1 to n)

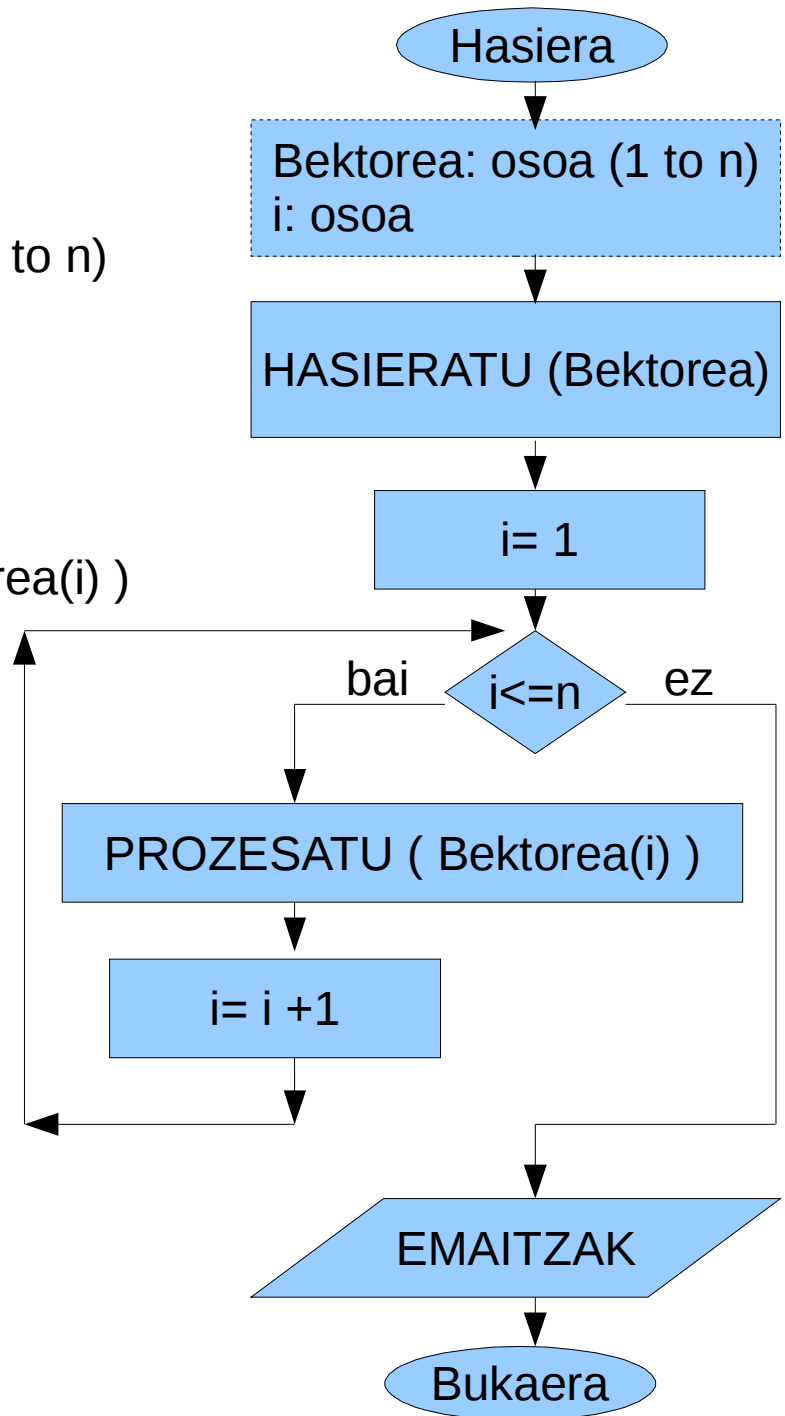
HASIERATU (Bektorea)

for i=1 to n

 PROZESATU (Bektorea(i))

next

EMAITZAK



3) Karaktere-kate bat:

```
Dim Katea as String
Dim karakterea as String
Dim i as Integer
Dim luzera as Integer
```

```
Katea= InputBox(...)
```

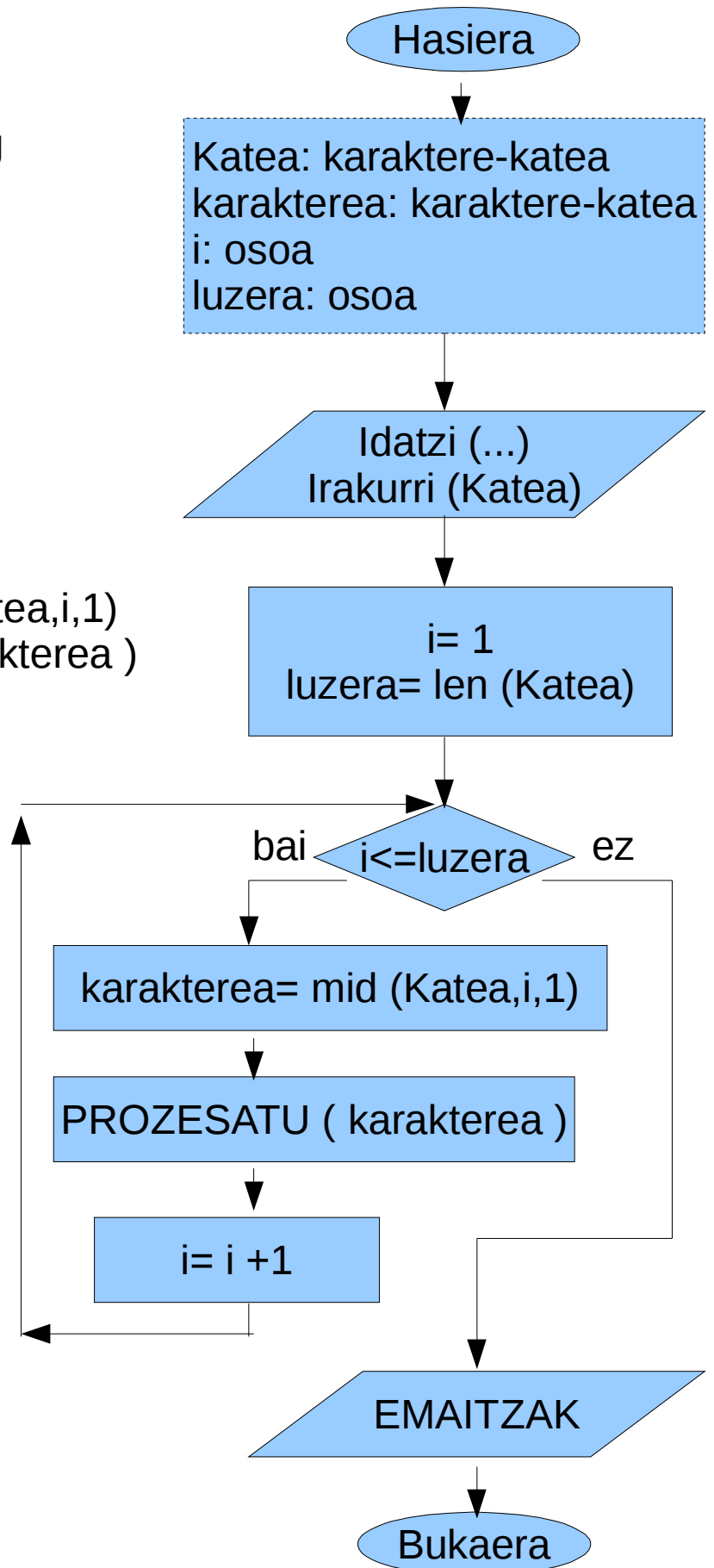
```
luzera= len(Katea)
```

```
for i=1 to luzera
```

```
    karakterea= mid(Katea,i,1)
    PROZESATU ( karakterea )
```

```
next
```

```
EMAITZAK
```



ADIBIDEAK

1) Zeroz bukatutako zenbaki sekuentzia bateko maximoa kalkulatu:

Dim x as Integer
Dim max as Integer

x= InputBox(...)
max= x

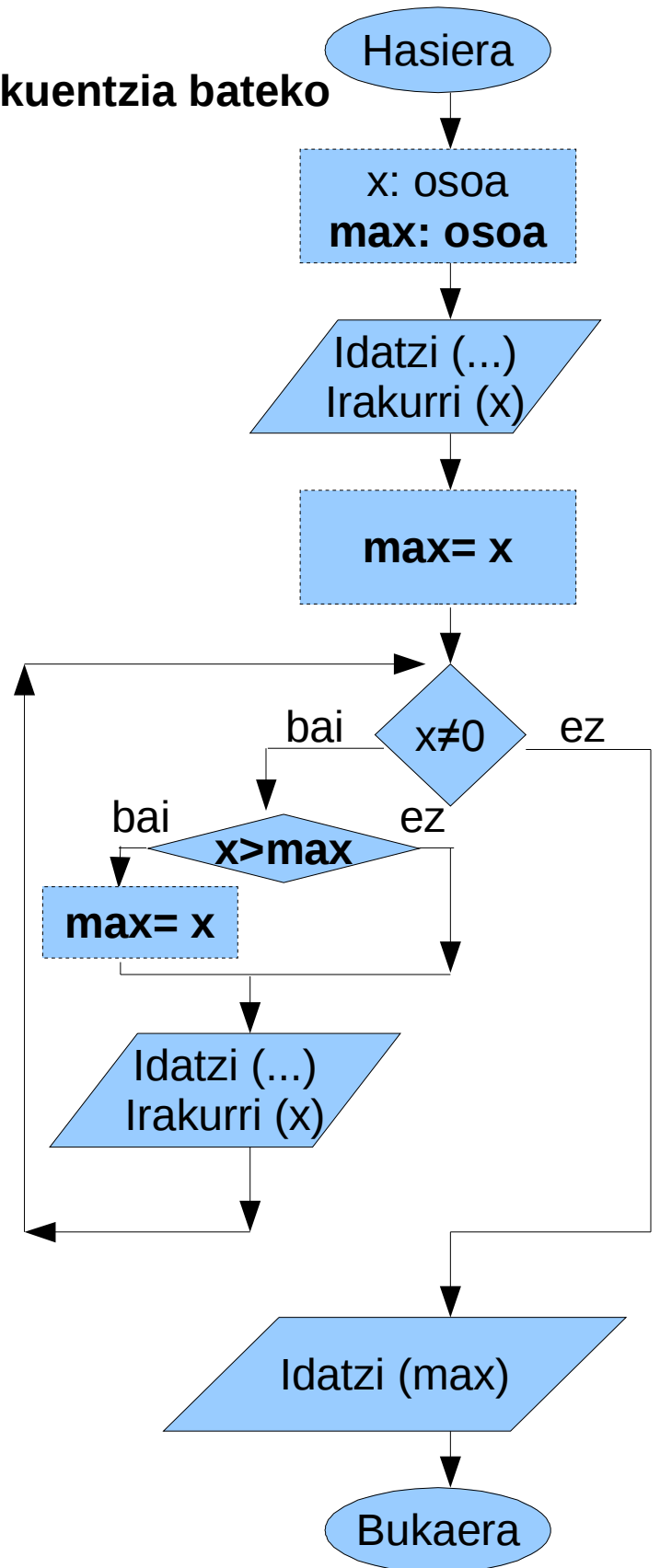
while x<>0

if x>max then
max= x
end if

x= InputBox(...)

wend

MsgBox(max)



2) Bektore bateko maximoa kalkulatu:

Dim x as Integer
Dim max as Integer

x= InputBox(...)
max= x

HASIERATU (Bektorea)

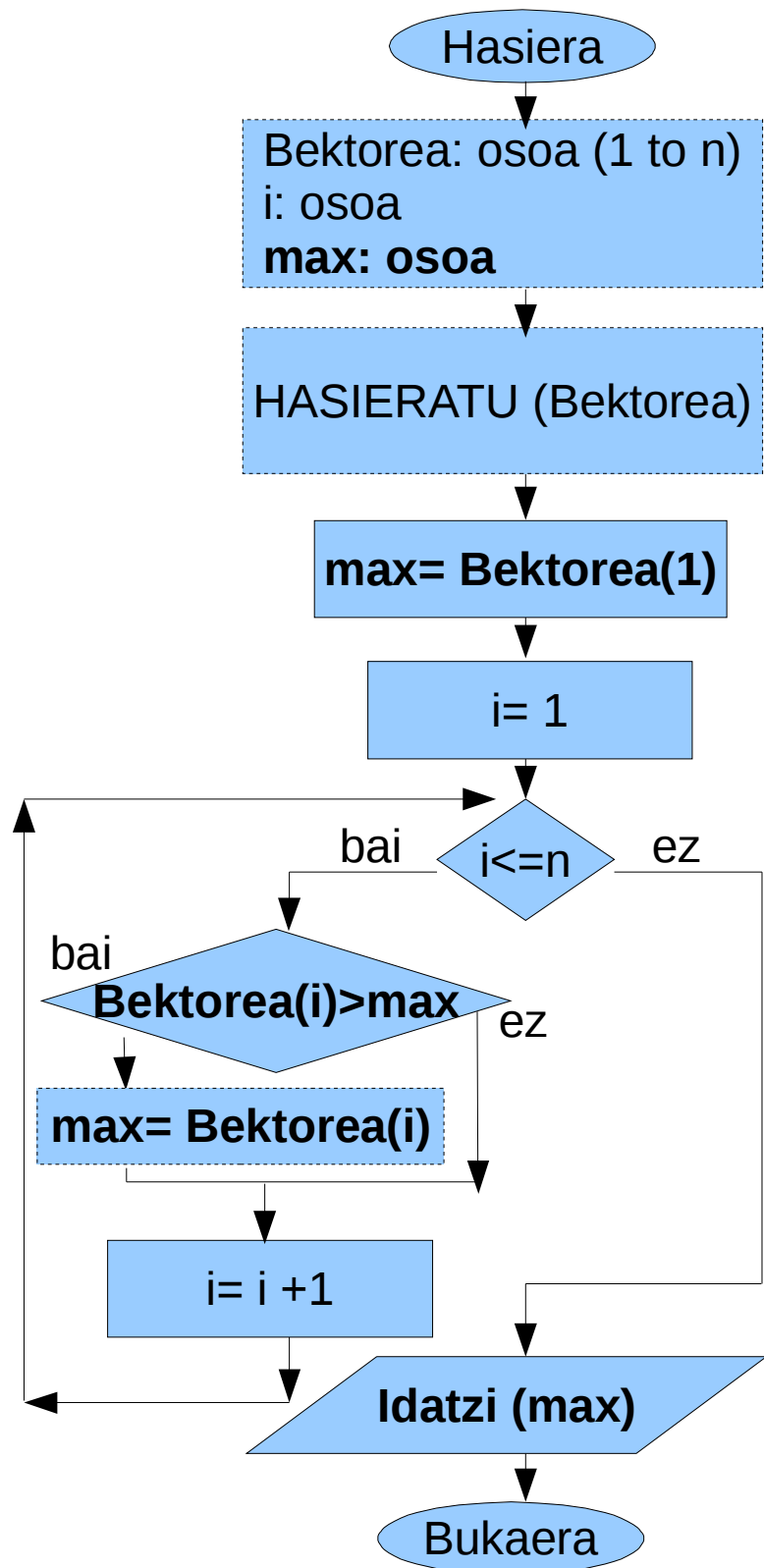
while x<>0

if x>max then
 max= x
end if

x= InputBox(...)

wend

MsgBox(max)



OINARRIZKO ALGORITMOAK:

1) Bektoreak

i) Batez-bestekoa

Dim A(1 to n) as Integer

Dim i as Integer

Dim batuketa as Integer

for i=1 to n

 batuketa= batuketa + A(i)

next

MsgBox(batuketa / n)

ii) Maximoa bilatu

Dim A(1 to n) as Integer

Dim i as Integer

Dim pos as Integer

max= A(1)

pos= 1

for i=2 to n

 if A(i)>max then

 max= A(i)

 pos= i 'bukaeran posizioa jakin nahi badugu

 end if

next

MsgBox ("Maximoa " & max & " da eta " & pos & ".
posizioan dago")

OINARRIZKO ALGORITMOAK:

iii) Sailkatu (handitik txikira)

Dim A(1 to n) as Integer

Dim i as Integer

Dim pos as Integer

for i=1 to n-1

 for j= i+1 to n

 if A(i)<A(j) then

 tmp= A(i)

 A(i)= A(j)

 A(j)= tmp 'A(i)<->A(j)

 end if

 next

next

AZPIPROGRAMAK (Funtzioak eta Prozedurak)

Zertarako balio dute? Errepikatzen den kodea behin bakarrik egin behar izateko.

1) FUNTZIOAK:

- Datu batzuk (parametroak) emanda emaitz bat itzultzen du.
- Itzuliko duen datu mota zehaztu behar da, bestela *Integer* bat itzuliko da.
- Emaitza itzultzeko funtzioaren izena aldagai bat balitz erabiltzen da.
- Funtzio bat erabiltzeko, bere izena eta pasa nahi dizkiogun datuak erabili behar

dira.

1.i)ADIBIDEA: Balio absolutua

'Lehenengo funtzioaren definizioa: parametro bat hartzen du (x, osoa) eta itzultzen 'duen balioa integer bat izango da

```
Private Function Abs (x As Integer) As Integer
```

```
    If x<0 then
```

```
        Abs= -x
```

```
    Else
```

```
        Abs= x
```

```
    End If
```

```
End Function
```

'Ondoren funtzioa erabiltzen duen botoi baten prozedura

```
Private Sub CommandButton1_Click()
```

```
    Dim zenbaki1 As Integer
```

```
    Dim zenbaki2 As Integer
```

```
    Dim emaitza As Integer
```

```
    zenbaki1= InputBox("Eman zenbaki bat")
```

```
    zenbaki2= InputBox("Eman beste bat")
```

```
    emaitza= Abs(zenbaki1) + Abs(zenbaki2)
```

'Abs funtzioari bietan deitzen zaio parametro ezberdinekin (zenbaki1 eta zenbaki2) 'eta dei bakoitza funtzioak itzulitako emaitzarekin ordezkutzen da

```
    MsgBox(emaitza)
```

```
End Sub
```

1.ii) ADIBIDEA: Batez-bestekoa

'Lehenengo funtzioaren definizioa: hiru parametro hartzen ditu (x, y eta z) eta itzultzen 'duen balioa erreal bat izango da

```
Private Function BB (x As Single, y As Single, z As Single) As Single
```

```
    BB= (x+y+z)/3
```

```
End Function
```

'Ondoren funtzioa erabiltzen duen botoi baten prozedura

```
Private Sub CommandButton1_Click()
```

```
    Dim zenbaki1 As Single
```

```
    Dim zenbaki2 As Single
```

```
    Dim zenbaki3 As Single
```

```
    zenbaki1= InputBox("Eman zenbaki bat")
```

```
    zenbaki2= InputBox("Eman beste bat")
```

```
    zenbaki3= InputBox("Eman beste bat")
```

```
    MsgBox( BB(zenbaki1,zenbaki2,zenbaki3) )
```

'Adibide honetan funtzioaren deia MsgBox-ari egindako dei baten barruan dago. Berdin 'berdin ordezkatzan da BBri egindako deia itzultzen duen emaitzarekin

```
End Sub
```

2) PROZEDURAK:

- Ez dute emaitzarik bueltatzen.
- Zertarako erabili? Adibidez, informazioa pantailaratzeko, bektore bat sailkatzeko, botoi bat sakatutakoan zeozer egiteko (CommandButton1_Click()),
- Prozedura bat erabiltzeko, **Call** agindua, prozeduraren izena eta pasa nahi dizkiogun datuak (parametroak) erabili behar dira.

2.i)ADIBIDEA: Bektore jakin bat pantailaratu.

Prozedura honek Bek izeneko bektorea pantailaratzen du

```
Dim Bek(1 to 100) As Integer
```

'Lehenengo prozeduraren definizioa: parametrorik ez du hartzen du

```
Private Sub PrintBek ()
```

```
    Dim i As Integer
```

```
    for i= 1 to 100
```

```
        Picture1.Print(Bek(i))
```

```
    next
```

```
End Sub
```

'Ondoren prozedura erabiltzen duen botoi baten prozedura

```
Private Sub CommandButton1_Click()
```

```
    Bek (2) = 15
```

```
    Call PrintBek()
```

```
    Bek (10) = 37
```

```
    Call PrintBek()
```

```
End Sub
```

2.ii)ADIBIDEA: Guk definitutako datu mota bat pantailaratu.

```
Private Type Lagun
```

```
    Izena As String
```

```
    Abizenak As String
```

```
    Adina As Integer
```

```
    Telefonoa As Long
```

```
End Type
```

```
Dim Bek(1 to 100) As Lagun
```

```
Dim lagunKop As Integer
```

```
'Lagun motako elementu bat emanda, bere informazioa pantailaratu
```

```
Private Sub PrintBek (x As Lagun)
```

```
    Picture1.Print ("Izena= " & x.Izena & " Abizenak=" & x.Abizena)
```

```
End Sub
```

```
'Bektoreko elementu guztiak pantailaratzeko erabiliko dugu
```

```
Private Sub CommandButton1_Click()
```

```
    Dim i As Integer
```

```
    for i=1 to lagunKop
```

```
        Call PrintLagun (Bek(i))
```

```
    next
```

```
End Sub
```

3) ByRef/ByVal:

Visual Basic-en azpiprograma bati parametroak pasatzeko (bai funtzioei, bai prozedureei) bi modu daude:

-ByVal (defektuzkoa): pasatako balioaren kopia bat pasatzen da.

-ByRef: pasatako aldagaiaren erreferentzi bat pasatzen da. Honek esan nahi du parametroari egindako aldaketa guztiak aldagaiari egingo zaizkiola

```
Private Function Adibidea (x As Integer, ByRef y As Integer) As Integer
```

```
    x= x+2
```

```
    y= y+2
```

```
    Adibidea= x+y
```

```
End Function
```

```
Private Sub CommandButton1_Click()
```

```
    Dim a,b,c As Integer
```

```
    a= 7
```

```
    b= 7
```

```
    c= Adibidea (a,b)
```

```
    MsgBox (a & "," & b & "," & c)
```

```
'Honek "7 9 18" pantailaratuko luke, x ByVal eta y ByRef bezala deklaratura daudelako
```

```
End Sub
```

4) BEKTOREAK:

2.ii adibidean, suposatu bektore bat baino gehiago pantailaratu nahi ditugula. Prozedura bakarra bektore ezberdinekin erabiltzeko, bektorea eta bere tamainua parametro bezala pasa behar dira. Parametroa bektore bat dela adierazteko, izena eta gero parentesiak jarriko zaizkio. Adibidez:

Private Sub PrintBek2(x() as Integer,...)

Orokorrean, bektoreekin prozedurak erabiliko ditugu (emaitz bakarra itzuli behar ez bada), eta bektore batean aldaketak egiteko ByRef deklaraturako da 2.ii adibidean bezala.

2.ii)ADIBIDEA: Edozein bektore pantailaratu.

Prozedura honek edozein tamainuko zenbaki osoko bektore bat pantailaratzen du

'Lehenengo prozeduraren definizioa:

'x bektorea izango da eta tamainua bere tamainua

Private Sub PrintBek2 (x() As Integer, tamainua as Integer)

Dim i As Integer

for i= 1 to tamainua

Picture1.Print(x(i))

next

End Sub

'Ondoren prozedura erabiltzen duen botoi baten prozedura

Private Sub CommandButton1_Click()

'Prozedurari bektorea pasatzen zaio, beraz bektoreak aldagai

'globalak edo lokalak izan litezke

Dim A(1 to 100) As Integer

Dim B(1 to 50) As Integer

Dim C(1 to 500) As Integer

Call PrintBek(A,100)

Call PrintBek(B,50)

Call PrintBek(C,500)

End Sub

Adibide honetan bi bektore beste batean batu nahi dira. Itzuli nahi den emaitza bektore bat denez, prozedura batean irteerako bektorea *ByRef* parametro bezala deklaratu da.

2.iii)ADIBIDEA: Bi bektore beste batean batu

'Lehenengo prozeduraren definizioa:

'x eta y sarreerako bektoreak izango dira, z irteerakoa eta n hiru bektoreen tamainua

```
Private Sub BatuBektoreak (x() As Integer, y() As Integer, ByRef z() As Integer, n As Integer)
    Dim i As Integer
    for i= 1 to n
        z(i)= x(i) + y(i)
    next
End Sub
```

'Ondoren prozedura erabiltzen duen botoi baten prozedura

```
Private Sub CommandButton1_Click()
    'Prozedurari bektorea pasatzen zaio, beraz bektoreak aldagai
    'globalak edo lokalak izan litezke
```

```
    Dim A(1 to 100) As Integer
    Dim B(1 to 100) As Integer
    Dim C(1 to 100) As Integer
```

'suposatu A eta B bektoreetan honezkero balioak ditugula

```
    Call BatuBektoreak(A,B,C)
    'C()= A() + B()
```

```
End Sub
```