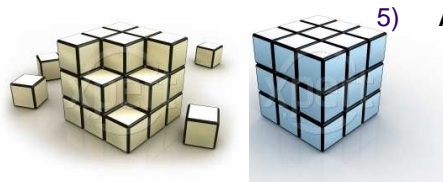


6-PROGRAMAZIO METODOLOGIA

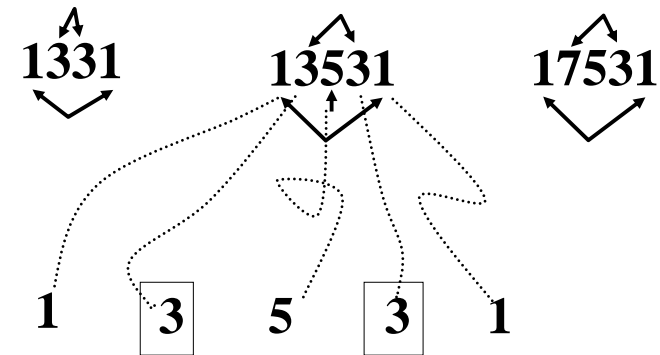
Programazio Modularra

- 1) Beharra
- 2) Abantailak
- 3) Adierazpidea
- 4) Adibideak
- 5) Azpi-algoritmoak



1.1 BEHARRA

Kapikua



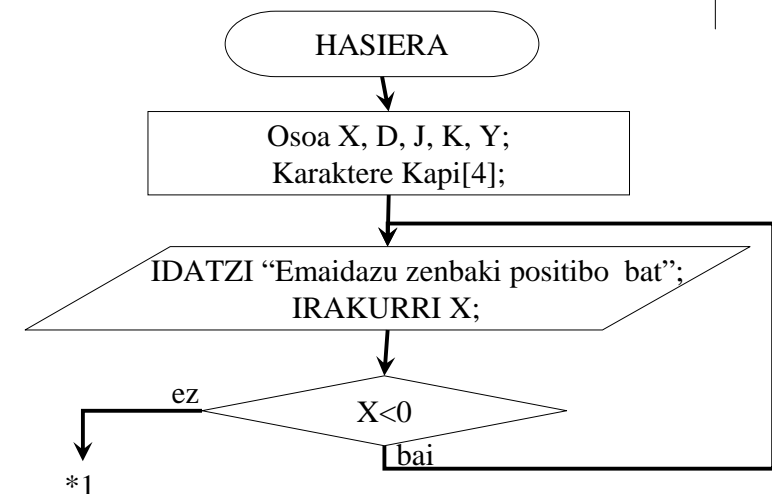
1.2 BEHARRA

Kapikua. Naturalki

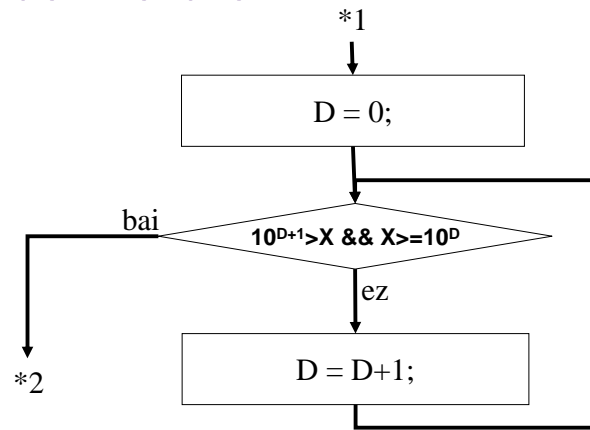
- Aurrebaldintza:
 - Zer da: **X: zenbakia**
 - Mota: **$X \in \mathbb{Z}$**
 - Baldintza indibidualak: **$X \Rightarrow 0$**
 - Erlazioak:
 - Ondorengo baldintza:
 - Zer da: **Kapi: mezu bat "BAI" ala "EZ" adieraziz**
 - Mota: **Kapi esaldia (karaktere Kapi[4])**
 - Baldintza indibidualak:
 - Erlazioak:
- Bada D, X-ren digitu kopurua ($10^{D+1} > X \geq 10^D$). J eta K-k hartu ditzatela jarraian $J = D .. 1$ eta $K = 1 .. D$ balioak.**
- J, K balio guztietan $(X \% 10^J) / 10^{J-1} == (X \% 10^K) / 10^{K-1}$
 - Kapi == "BAI"
 - Bestela
 - Kapi == "EZ"

1.3 BEHARRA

Kapikua. Naturalki



1.4 BEHARRA Kapikua. Naturalki

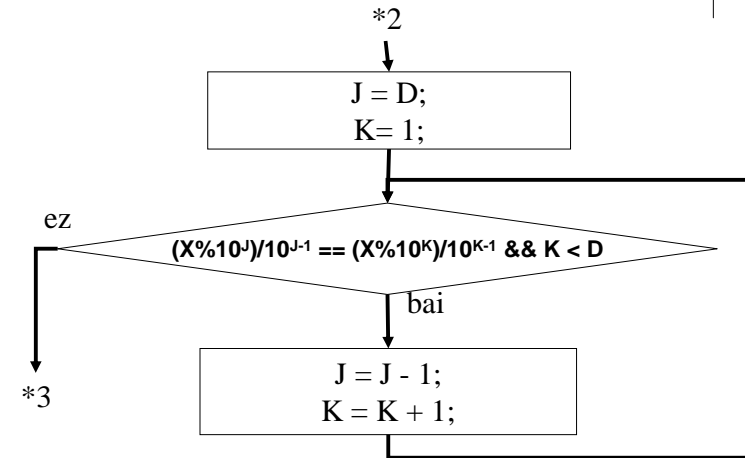


2007-2008

Informatikaren Oinarriak - Iker Azpeitia

5

1.5 BEHARRA Kapikua. Naturalki

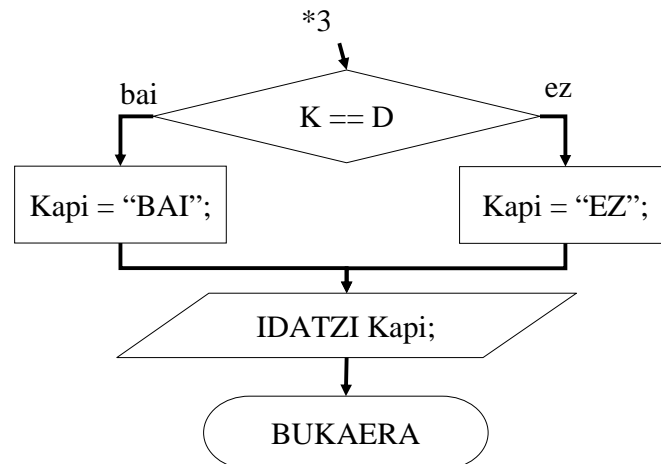


2007-2008

Informatikaren Oinarriak - Iker Azpeitia

6

1.6 BEHARRA Kapikua. Naturalki



2007-2008

Informatikaren Oinarriak - Iker Azpeitia

7

1.7 BEHARRA Murrizpenak



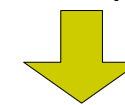
- Algoritmoetan eragile matematiko hauek erabili ditzakegu:

- -, +, *, /, %

- Ezin ditugu berreketak erabili

- $10^{D+1} > X \ \&\& \ X \geq 10^D$

- $(X \% 10^J) / 10^{J-1} == (X \% 10^K) / 10^{K-1}$



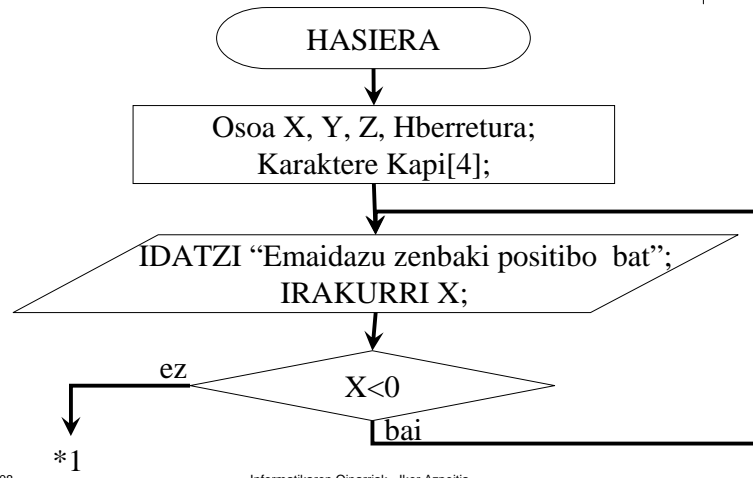
- Algoritmoa aldatu egin behar da

2007-2008

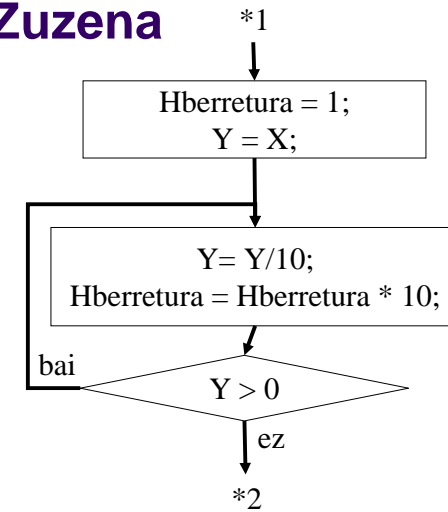
Informatikaren Oinarriak - Iker Azpeitia

8

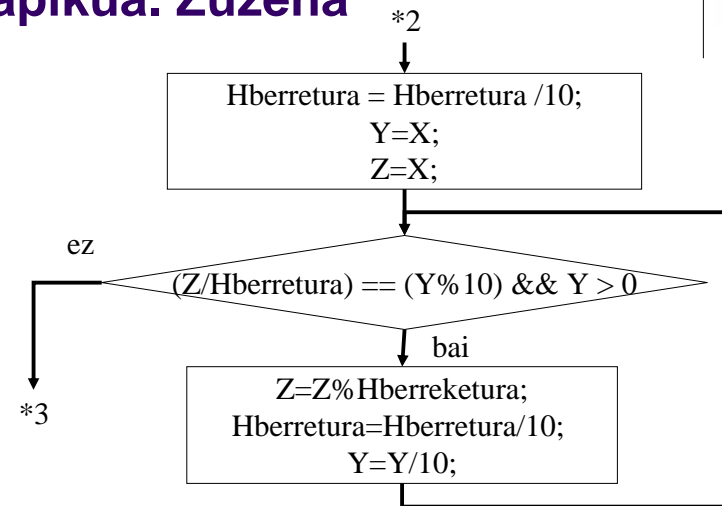
1.8 BEHARRA Kapikua. Zuzena



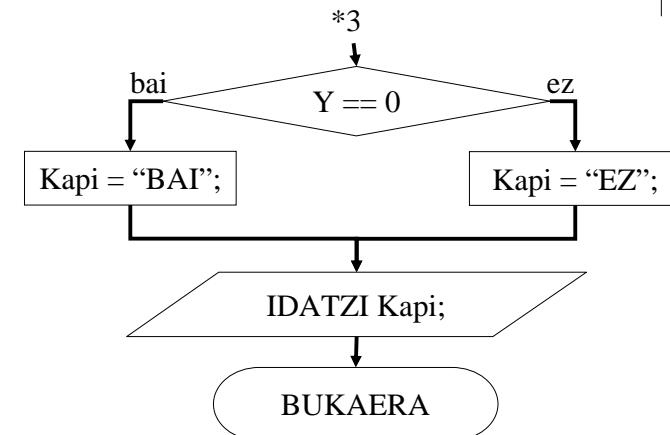
1.9 BEHARRA Kapikua. Zuzena



1.10 BEHARRA Kapikua. Zuzena



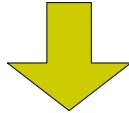
1.11 BEHARRA Kapikua. Zuzena



1.12 BEHARRA Kontraesana

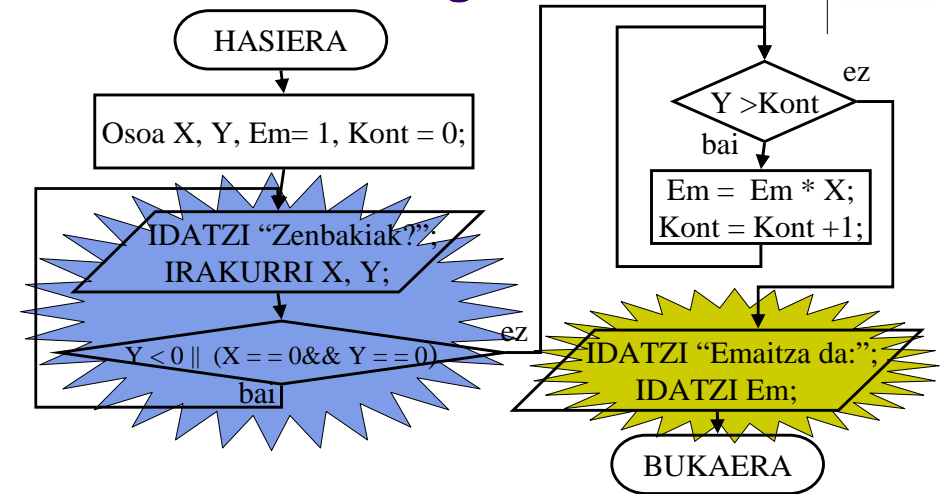


- Berreketa ebazten duen algoritmoa sortu badugu, ezin al dugu berrerabili?

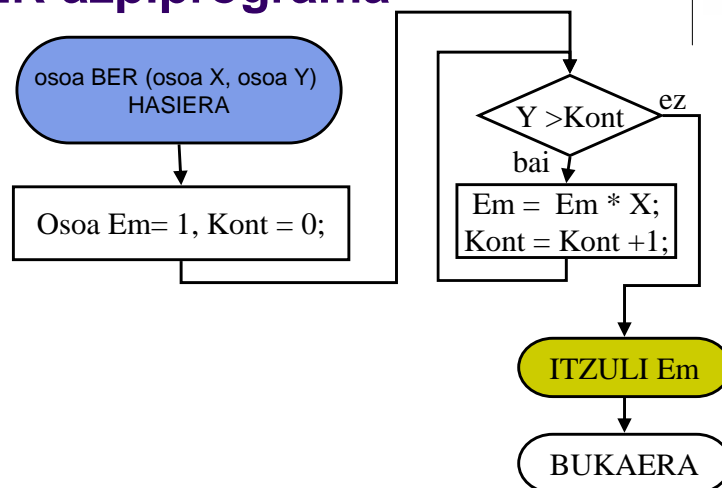


- Bai. Zenbait gauza aldatuz.
- Datu sarrera eta datu irteera. Programen arteko datu trukaketa.

1.13 BEHARRA BERREKETA algoritmoa



1.14 BEHARRA BER azpiprograma



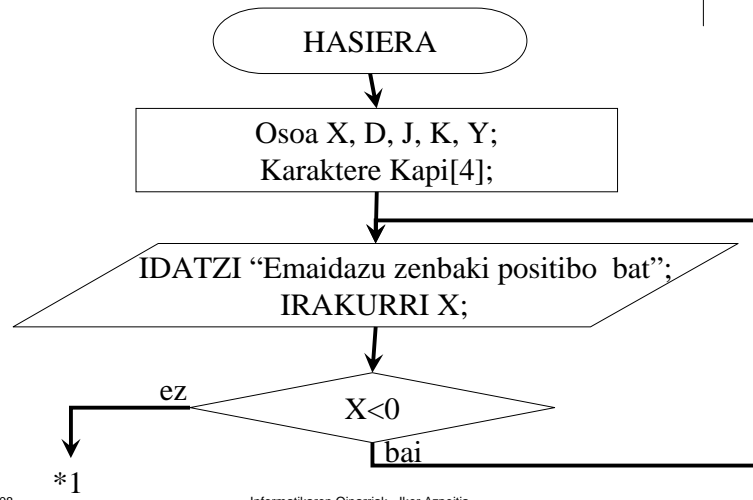
1.15 BEHARRA Kapikua. BER azpiprogramaz



- Aurrebaldintza:
 - Zer da: **X: zenbakia**
 - Mota: **$X \in \mathbb{Z}$**
 - Baldintza indibidualak: **$X \Rightarrow 0$**
 - Erlazioak:
 - Ondorengo baldintza:
 - Zer da: **Kapi: mezu bat "BAI" ala "EZ" adieraziz**
 - Mota: **Kapi esaldia (karaktere Kapi[4])**
 - Baldintza indibidualak:
 - Erlazioak:
- Bada D, X-ren digitu kopurua ($10^{D+1} > X \geq 10^D$). J eta K-k hartu ditzatela jarraian $J = D .. 1$ eta $K = 1 .. D$ balioak.**
- J, K balio guztietan $(X \% 10^J) / 10^{J-1} == (X \% 10^K) / 10^{K-1}$
 - Kapi == "BAI"
 - Kapi == "EZ"
 - Bestela

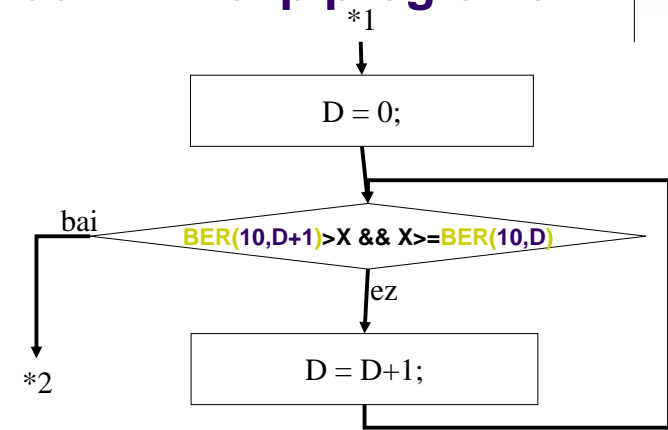
1.16 BEHARRA

Kapikua. BER azpiprogramaz



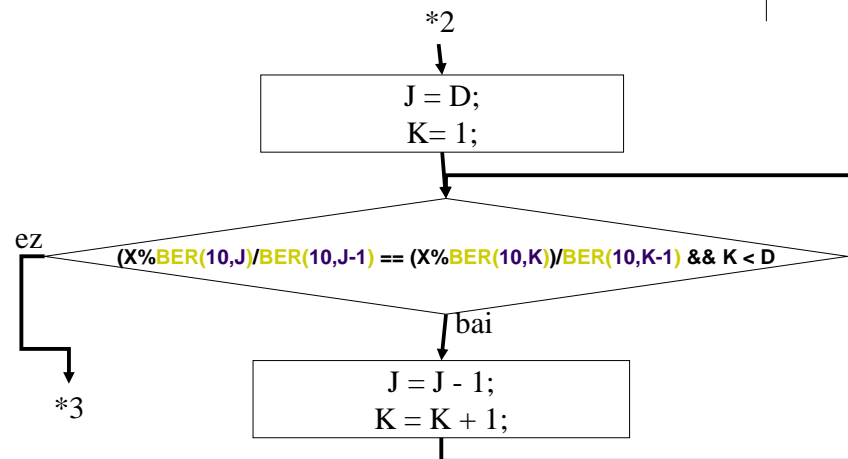
1.17 BEHARRA

Kapikua. BER azpiprogramaz



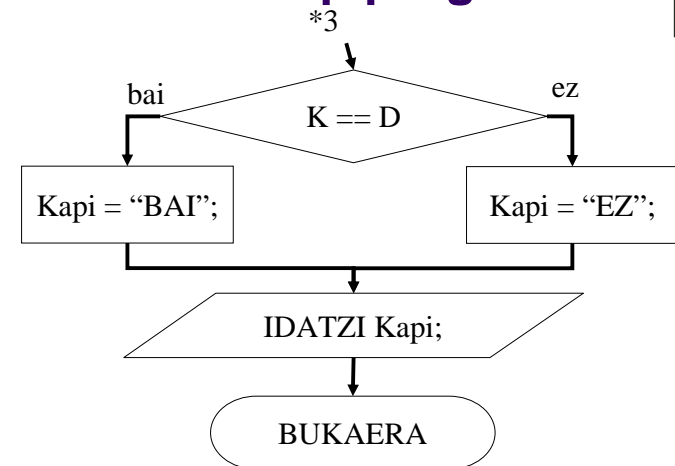
1.18 BEHARRA

Kapikua. BER azpiprogramaz



1.19 BEHARRA

Kapikua. BER azpiprogramaz

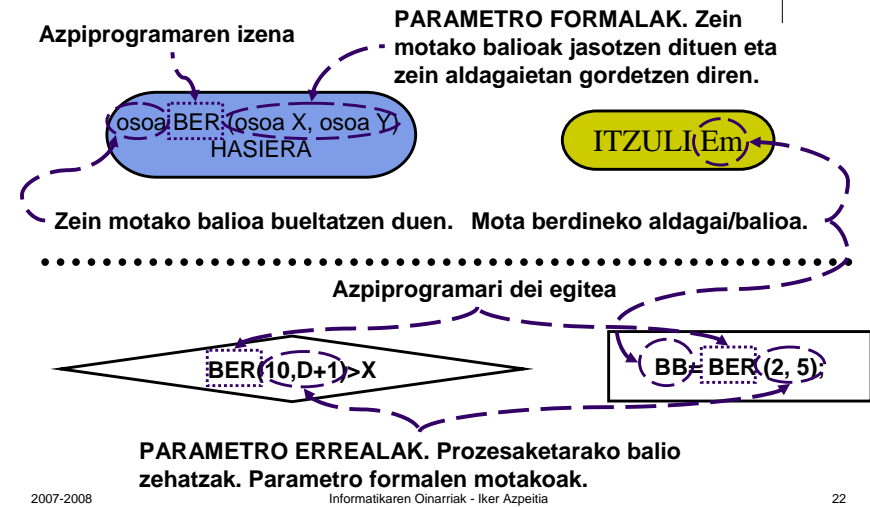


2.1 ABANTAILAK

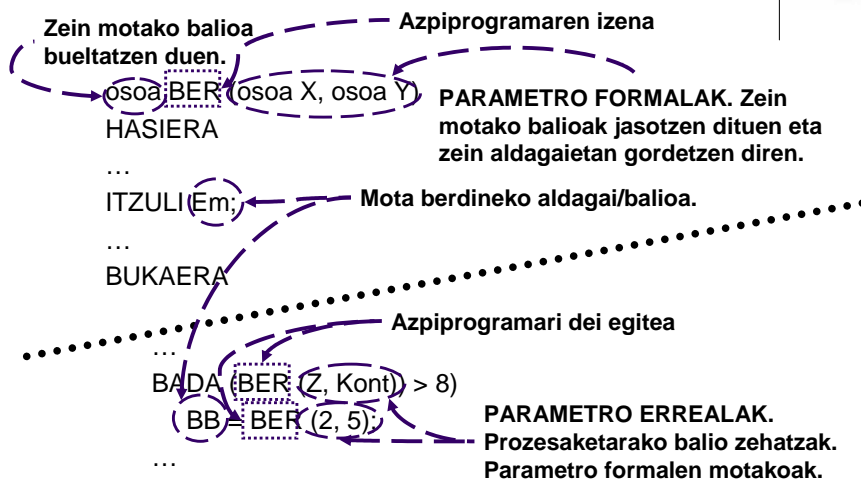


- Arazo konplexuak zatituz erraztu
 - Abstrakzioaz ulergarritasuna lortu
 - Zati bakoitza independente sortu eta probatu
- Azpiprogramak berrerabili
 - Denbora aurreztu
 - Akats gutxiago
 - Parametroen bitartez ebazpenak orokortu
 - Berbidura → **BER (2, ?)**
 - $(2^3)^4$ → **BER(BER (2,3), 4)**

3.1 ADIERAZPIDEA Fluxu diagraman



3.2 ADIERAZPIDEA Sasikodean



4.1 ADIBIDEAK Posizioa azpi-algoritmoa



- Esaldi baten non agertzen den karaktere zehatz bat
- Zehaztapena:
 - Aurrebaldintza:
 - Zer da:
 - erak: karaktere kateari erakuslea
 - kar: aurkitu beharreko karakterea
 - topea: gehienez zenbat posizio bilatu
 - Mota:
 - karakterea *erak;
 - karakterea kar;
 - osoa topea; topea ∈ Z
 - Baldintza indibidualak: topea ≥ 0
 - Erlazioak:

4.2 ADIBIDEAK

Posizioa azpi-algoritmoa

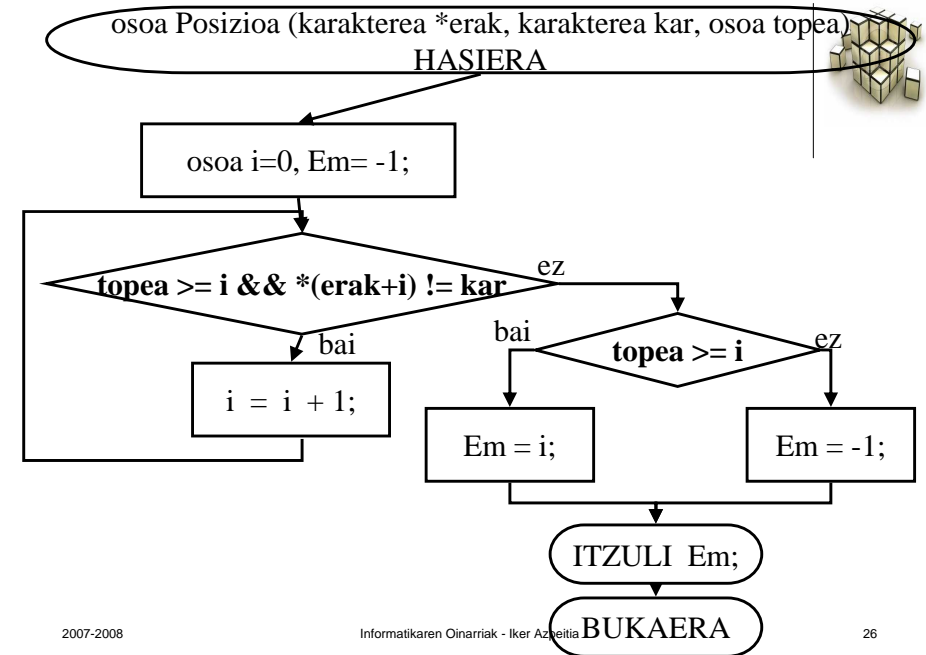


- Zehaztapena:
 - Ondorengo baldintza:
 - Zer da: **Em**: zein posiziotan dagoen kar *erak apuntatzen duen karaktere katean
 - Mota: **Em** $\in \mathbb{Z}$
 - Baldintza indibidualak: **Em** ≥ -1
 - Erlazioak:
 - Existitzen bada **i** balioa betetzen duena ***(erak+i) == kar** eta **i** ≥ 0 eta **i** \leq **Topea** eta beste edozein **j** balioak betetzen badu **0** \leq **j** $<$ **i** eta ***(erak+i) != kar** \rightarrow **Em == i**
 - Bestela \rightarrow **Em == -1**

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

25



2007-2008

Informatikaren Oinarriak - Iker Azpeitia

26

4.4 ADIBIDEAK

Kalkulagailua



- Bi zenbaki osoen kalkulagailu bat (batu, kendu eta biderkatu)
 - Kalkulagailuaren eragiketa bakoitza azpi-problema bat
 - Programa nagusia beste bat.

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

27

4.5 ADIBIDEAK

Kalkulagailua. Algoritmo nagusia



hutsa Kalkulagailua ()

HASIERA

osoa Auk, X, Y, Z;

EGIN

1 IDATZI ("1. Batu; 2. Kendu; 3. Biderkatu; 4. Irten");

2 IDATZI (" Zer egin nahi duzu?");

3 IRAKURRI Auk;

4 DENBITARTEAN (Auk < 1 || Auk > 4);

5 IDATZI ("Ze bi zenbaki osoekin?");

6 IRAKURRI X, Y;

BALDIN (Auk)

7 BADA 1: Z = Batura (X, Y); 8

IDATZI Z; 9

IRTEN; 10

11 BADA 2: Z = Kendura (X, Y); 12

IDATZI Z; 13

IRTEN; 14

15 BADA 3: Z = Biderkadura (X, Y); 16

IDATZI Z; 17

IRTEN; 18

19 BESTELA: IDATZI "Eskerrik asko kalkulagailua erabiltzeagatik"; 20

BUK_BAILDIN;

DENBITARTEAN (Auk != 4); 21

BUKAERA

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

28

4.6 ADIBIDEAK

Kalkulagailua. Batura algoritmoa



1 osoa Batura (osoa A, osoa B)

HASIERA

osoa Em;

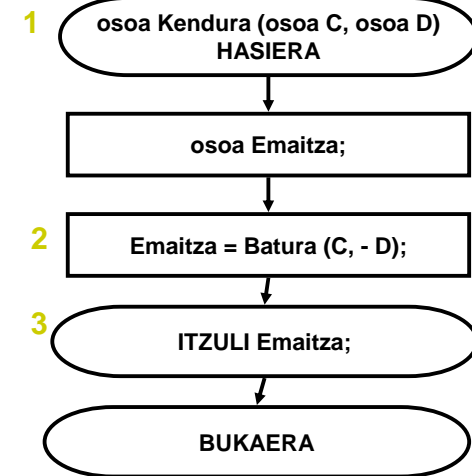
2 $Em = A + B$;

3 ITZULI Em;

BUKAERA

4.7 ADIBIDEAK

Kalkulagailua. Kendura algoritmoa



4.8 ADIBIDEAK

Kalkulagailua. Biderkadura algoritmoa



1 osoa Biderkadura (osoa F, osoa G)

HASIERA

2 Osoa $Em = 0$;

3 BADA ($F < 0$)

4 $F = F * -1$;

5 $G = G * -1$;

BUK_BADA

6 DENBITARTEAN ($F > 0$)

7 $Em = \text{Batura}(Em, G)$;

8 $F = F - 1$;

BUK_DENBITARTEAN

9 ITZULI Em;

BUKAERA

5.1 EXEKUZIOA

Kalkulagailua. Simulazio taula



- Erabiltzaileak -4, 1, 2, 3, 4, 5 eta 6 sakatzen du.

#	Auk	X	Y	Z	Batura A	Batura B	Batura Em	Pant.
1	?	?	?	?				1. ...
2	?	?	?	?				Zer ...
3	-4	?	?	?				
4	-4	?	?	?				
1	-4	?	?	?				1. ...
2	-4	?	?	?				Zer ...

5.2 EXEKUZIOA

Kalkulagailua. Simulazio taula



#	Auk	X	Y	Z	Batura A	Batura B	Batura Em	Pant.
3	1	?	?	?				
4	1	?	?	?				
5	1	?	?	?				Ze bi.
6	1	2	3	?				
7	1	2	3	?				
8	8	1	2	3	?			
8	Batura 1	1	2	3	?	2	3	?

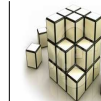
2007-2008

Informatikaren Oinarriak - Iker Azpeitia

33

5.3 EXEKUZIOA

Kalkulagailua. Simulazio taula



#	Auk	X	Y	Z	Batura A	Batura B	Batura Em	Pant.
8	Batura 2	1	2	3	?	2	3	5
8	Batura 3	1	2	3	?	2	3	5
8	8	1	2	3	5			
9	1	2	3	5				5
10	1	2	3	5				
21	1	2	3	5				
1	1	2	3	5				1. ...

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

34

5.4 EXEKUZIOA

Kalkulagailua. Simulazio taula



#	Auk	X	Y	Z	Batura A	Batura B	Batura Em	Pant.
2	1	2	3	5				Zer ...
3	4	2	3	5				
4	4	2	3	5				
5	4	2	3	5				Ze bi.
6	4	5	6	5				
7	4	5	6	5				
11	4	5	6	5				

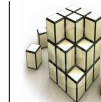
2007-2008

Informatikaren Oinarriak - Iker Azpeitia

35

5.5 EXEKUZIOA

Kalkulagailua. Simulazio taula



#	Auk	X	Y	Z	Batura A	Batura B	Batura Em	Pant.
15	4	5	6	5				
19	4	5	6	5				
20	4	5	6	5				Esk...
21	4	5	6	5				

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

36

5.6 EXEKUZIOA

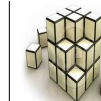
Simulazio taulak



- Abantaila: “Zati bakoitza independente sortu eta probatu”
 - Azpi-algoritmoa bere aldetik probatu. Itxuraz ondo badabil ...
- ↓
- ...ez dugu beste algoritmoen simulazio taulan aginduz-agindu exekutatu behar.
 - Probatu ondoren, ondo dabilela suposatu eta emaitza erabili.

5.7 EXEKUZIOA

Biderkadura. Simulazio taula



- Kasu orokorra: Biderkadura $(-2, 3) == -6$

#	F	G	Em	Pant.
1	-2	3	?	
2	-2	3	0	
3	-2	3	0	
4	2	3	0	
5	2	-3	0	
6	2	-3	0	

5.8 EXEKUZIOA

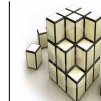
Biderkadura. Simulazio taula



#	F	G	Em	Pant.
7	2	-3	-3	
8	1	-3	-3	
6	1	-3	-3	
7	1	-3	-6	
8	0	-3	-6	
6	0	-3	-6	
9	0	-3	-6	

5.9 EXEKUZIOA

Biderkadura. Simulazio taula



- Kasu orokorra: Biderkadura $(-2, -3) == 6$

#	F	G	Em	Pant.
1	-2	-3	?	
2	-2	-3	0	
3	-2	-3	0	
4	2	-3	0	
5	2	3	0	
6	2	3	0	

5.10 EXEKUZIOA

Biderkadura. Simulazio taula



#	F	G	Em	Pant.
7	2	3	3	
8	1	3	3	
6	1	3	3	
7	1	3	6	
8	0	3	6	
6	0	3	6	
9	0	3	6	

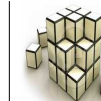
2007-2008

Informatikaren Oinarriak - Iker Azpeitia

41

5.11 EXEKUZIOA

Biderkadura. Simulazio taula



- Kasu orokorra: Biderkadura (2, -3) == -6

#	F	G	Em	Pant.
1	2	-3	?	
2	2	-3	0	
3	2	-3	0	
6	2	-3	0	

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

42

5.12 EXEKUZIOA

Biderkadura. Simulazio taula



#	F	G	Em	Pant.
7	2	-3	-3	
8	1	-3	-3	
6	1	-3	-3	
7	1	-3	-6	
8	0	-3	-6	
6	0	-3	-6	
9	0	-3	-6	

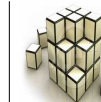
2007-2008

Informatikaren Oinarriak - Iker Azpeitia

43

5.13 EXEKUZIOA

Biderkadura. Simulazio taula



- Kasu orokorra: Biderkadura (2, 3) == 6

#	F	G	Em	Pant.
1	2	3	?	
2	2	3	0	
3	2	3	0	
6	2	3	0	

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

44

5.14 EXEKUZIOA

Biderkadura. Simulazio taula



#	F	G	Em	Pant.
7	2	3	3	
8	1	3	3	
6	1	3	3	
7	1	3	6	
8	0	3	6	
6	0	3	6	
9	0	3	6	

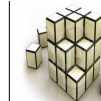
2007-2008

Informatikaren Oinarriak - Iker Azpeitia

45

5.15 EXEKUZIOA

Biderkadura. Simulazio taula



- Kasu berezia: Biderkadura $(0, 3) == 0$

#	F	G	Em	Pant.
1	0	3	?	
2	0	3	0	
3	0	3	0	
6	0	3	0	
9	0	3	0	

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

46

5.16 EXEKUZIOA

Biderkadura. Simulazio taula



- Kasu berezia: Biderkadura $(-2, 0) == 0$

#	F	G	Em	Pant.
1	-2	0	?	
2	-2	0	0	
3	-2	0	0	
4	2	0	0	
5	2	0	0	
6	2	0	0	

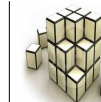
2007-2008

Informatikaren Oinarriak - Iker Azpeitia

47

5.17 EXEKUZIOA

Biderkadura. Simulazio taula



#	F	G	Em	Pant.
7	2	0	0	
8	1	0	0	
6	1	0	0	
7	1	0	0	
8	0	0	0	
6	0	0	0	
9	0	0	0	

2007-2008

Informatikaren Oinarriak - Iker Azpeitia

48

5.18 EXEKUZIOA

Biderkadura. Simulazio taulak



- Kasu bereziak eta orokor batzuk probatuta ondo dabilela dirudi...
- ...suposatuz *Batura* azpialgoritmoa ondo dabilela
- *Biderkadura* azpialgoritmoa erabiltzen duten algoritmoek emaitza egokia bueltatzen duela suposatu dezakete