



Oinarrizko Programazioa

2. Programazioko oinarrizko kontzeptuak



Edukiak

1. Sarrera
2. **Programazioko oinarrizko kontzeptuak**
3. Programen beheranzko diseinua
4. Oinarrizko datu-egiturak
5. Programazio-lengoaiei erabilera
6. Aplikazio-adibideak



Algoritmoen elementuak

- Objektuak
 - Informazioaren errepresentazioa
 - Objektu sinpleak eta egituratuak
- Adierazpenak (espresioak)
- Objektuak erabiltzeko oinarrizko aginduak
 - Datuak irakurri
 - Datuak idatzi
 - Asignazioa
- Kontrol-egiturak
 - Baldintzazko egiturak
 - Iterazio-egiturak
- Moduluak

Oinarrizko elementuak asignaturan

- Objektuak.
 - Objektu sinpleak → 2.1. Informazioaren errepresentazioa
 - Objektu egituratuak → 2.2. Adierazpenak.
- Adierazpenak → 2.3. Aginduak.
- Objektuak erabiltzeko oinarrizko aginduak → 2.4. Kontrol-egiturak.
- Kontrol-egiturak → 3. Azpiprogramak.
- Moduluak → 4. Oinarrizko datu-egiturak



Oinarrizko Programazioa

2. Programazioko oinarrizko kontzeptuak.

2.1. Sarrera

2.2. Informazioaren errepresentazioa.

Objektu konstanteak eta aldagaiak.

Oinarrizko datu-motak.

2.3. Adierazpenak.

2.4. Datuak manipulatzeko oinarrizko aginduak.

2.5. Kontrol-egiturak.

Baldintzazko egiturak

Iterazioa. Inbariantearen kontzeptua

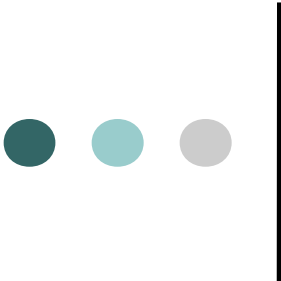
2.6. Algoritmoen idazkera. Oinarrizko eskemak.



2.2. Objektuak.

Informazioaren errepresentazioa

- Zerbait behar dugu datuak errepresentatzeko
- Algoritmoak balio behar du exekuzio desberdinetan datu desberdinekin erabilia izateko. Adibidez: zkh
- Aldagaia:
 - kutxa bat balio bat uzteko balio duena
 - balio hori ikusi ahal izango da gero
 - Balio berri bat jar daiteke (asignazioa), baina horrelakoetan lehengo balioa galdu egiten da.
- Balio sinpleak (letra bat, zenbaki bat, ...) eta balio egituratuak (persona bat: izena,deiturak, adina,...)



Objektuak erabiltzeko oinarrizko aginduak

- Datuak irakurri
 - Datuen balioak sarreratik eskuratzeko
- Datuak idatzi
 - Datuen balioak eta mezuak erakusteko
- Asignazioa
 - Datuen balioak manipulatzeko:
eragiketa aritmetikoak, balioak kopiatzeko...



Adierazpenak

Kalkulatu behar diren espresioak adierazteko

$$\frac{Ax^2+3}{Bx-1}$$

$$A*(X*X)+3 / B*X-1 \quad ?$$



Kontrol-egiturak

- Programak agindu-sekuentziak dira.
- Programa exekutatzea bere aginduak banan-banan exekutatzea da, sekuentziako ordena errespetatuz.
- Baina aldagaietan dauden balioak erabiliz baldintza bat (adierazpen boolear bat) betetzen bada, posible da aginduen exekuzio sekuentziala aldatzea:
 - Agindu multzo bat egin edo ez.
 - Agindu multzo bat hainbatetan errepikatzea.
 - Agindu multzo bat errepikatu etengabe baldintza bete arte.



Moduluak

- Algoritmoak oso konplexuak izan daitezke
 - ulertzeko, konpontzeko edo berrerabiltzeko
- Moduluak, azpialgoritmoak, errutinak, klaseak:
 - logikoki independenteak diren datu- eta agindu-multzotan banatzeko erabiltzen dira
- Lana banatu ahal izango da horrela



Informazioaren errepresentazioa

- Konstanteak eta aldagaiak
- Literalak
- Oinarrizko datu-motak:
 - zenbaki osoak, zenbaki errealak, karaktereak, boolearrak



Objektuak

- Datu-objektu bakoitzak hiru ezaugarri ditu:
 - IZENA: identifikadore bat
 - MOTA: Mota batek definitzen du:
 - bere balio posibleen multzoa.
 - balio horiekin egin daitezkeen eragiketak.
 - Adibidez: osokoa, errealak, karakterea.
- BALIOA: programaren exekuzioko une zehatz batean daukana



Konstanteak eta aldagaiak

Objektu konstantea (konstantea)		Objektu aldagaia (aldagaia)	
Izena	aldaeza	Izena	aldaeza
Mota	aldaeza	Mota	aldaeza
Balioa	aldaeza	Balioa	aldagaia



Konstanteak eta aldagaiak

- **Konstante batean**, programaren exekuzioan behin bere balioa finkatu denetik aurrera, **ezin izango zaio balio hori aldatu objektuari**.
 - Adibidez:
 - `Pi` `3.14159`
 - `Seg_Orduko` `3600`
- **Aldagai batean**, berriz, **uneko balioa alda daiteke exekuzioan zehar nahi beste aldiz**.
 - Adibidez:
 - `Kontagailua` `21`
 - `Urtea` `2014`



Literalak

- Noski, balioa bera ere erabil daiteke programetan. Halakoetan balioari “literal” esaten zaio.
 - Ez dago beti identifikadore bat erabili beharrik.
 - Adibidez:
 - Osoko literalak: 0 1 60 1000000
 - Literal errealak: 0.0 3.14158
 - Karaktereak: 'H' ':' ''
 - Karaktere-kateak: "Ordua: " "???"



Identifikadoreak

- Konstante, aldagai, mota, algoritmo eta programetako beste entitate batzuei ematen zaizkien izenak dira.
- Horrelako entitate bat identifikadore batekin lotu ondoren, aurrerantzean identifikadore hori erabiliko da programan entitate hori aipatzeko.



Programazio-estiloa

Iruzkinak eta identifikadore egokiak

- Komeni da iruzkinak erabiltzea programaren eginkizuna adierazteko eta zati bakoitzaren funtzionamendua esplikatzeko.
 - Iruzkinak adan: "--" ondoren lerro bukaeraraino
- Identifikadore ahalik eta deskriptiboenak aukeratu behar dira.



Oinarrizko DATU-MOTAK

- Balio bakarra hartzen dutenak dira oinarrizkoak:

Datu-mota

Balioak

Ada-z

Osokoa

Zenbaki osoak

Integer

Erreala

Zenbaki errealak

Float

Karakterea

Karakterearak

Character

Boolearra

Balio boolearrak

Boolean

Katea

Karaktere-kateak

String



Osoko datu-mota (*Integer*)

- Zenbaki osoen datu-mota
 - Domeinua:
..., -3, -2, -1, 0, +1, +2, +3, ...
 - Erne! Errepresentazio bitarra dela eta mugak ditu goitik eta behetik
 - 16 bitekin: [-32768,+32767]
 - Adaz: **[Integer'First, Integer'Last]**



Osoko datu-mota

- Eragiketak:

- Eragigaiak eta emaitzak osokoa motakoak dira derrigorrez.
- **Adi zatiketarekin!**

Eragile diadikoak

+	batuketa
-	kenketa
*	biderkaketa
/	zatiketa moztua
rem	zatiketaren hondarra
mod	modulua
**	berreketa

Eragile monadikoak

abs	balio absolutua
-	ukapena

eta gainera eragiketa erlazionalak: <, <=, >, >=, /=



Erreal datu-mota

- Zenbaki errealak adierazteko datu-mota

- Domeinua:

0.0 1.5 3.86473e5 3.0e+8 0.1234E-20

- Eragiketak:

- Eragigaiak eta emaitzak zenbaki errealak motakoak dira.

Eragile diadikoak

+ batuketa

- kenketa

* biderkaketa

/ zatiketa

** berreketa (berretzailea osoko motakoa)

eta gainera eragiketa erlazionalak: <, <=, >, >=, /=

Eragile monadikoak

abs balio absolutua

- ukapena



Boolear datu-mota

- Domeinua: {true, false} (egiazkoa eta faltsua)
- Eragiketak:
 - Eragile erlazionalak.
 - eragigai biak mota berekoak dira
 - emaitza boolearra.

=	berdin	<=	txikiago edo berdin
/=	desberdin	>=	handiago edo berdin
<	txikiago	>	handiago
 - Eragile logikoak.
 - eragigai biak eta emaitza boolearrak dira.
and (eta) , or (edo), xor (edo eskusiboa), not (ez)



Eragiketa boolearren egi-taulak

A	B	A eta B A and B	A edo B A or B	A ala B A xor B	ez B not B
False	False	False	False	False	True
False	True	False	True	True	False
True	False	False	True	True	
True	True	True	True	False	



Karaktere datu-mota

- Karaktereak adierazteko datu-mota
 - Domeinua:
 - ISO standarreko 128 karaktereak
 - 95 grafikoak: 'a' 'A' '1' '?' ...
 - 33 kontrolekoak: ff cr ...
 - **Kontuz ! 3 eta '3' balio desberdinak dira.**
 - Eragiketak:
 - Eragiketa erlazionalak: <, <=, >, >=, /=, =

Karaktere datu-mota (*Character*)

	0	1	2	3	4	5	6	7
0	NULL	SOH	STX	ETX	EOT	ENQ	ACK	BEL
8	BS	HT	LF	VT	FF	CR	SO	SI
16	DLE	DC1	DC2	DC3	DC4	NAC	SYN	ETB
24	CAN	EM	SUB	ESC	FS	GS	RS	US
32	SP	!	"	#	\$	%	&	'
40	()	*	+	,	-	.	/
48	0	1	2	3	4	5	6	7
56	8	9	:	;	<	=	>	?
64	@	A	B	C	D	E	F	G
72	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W
88	X	Y	Z	[\]	^	_
96	`	a	b	c	d	e	f	g
104	h	i	j	k	l	m	n	o
112	p	q	r	s	t	u	v	w
120	x	y	z	{		}	~	DEL



Kate datu-mota (string)

- Karaktere-kateak adierazteko datu-mota
 - Domeinua:
 - Karaktere-kateak komatxoaren artean
"Maite"
"Kale Nagusia 12, 3.C"
 - Kontuz! "3", 3 eta '3' balio desberdinak dira.
 - Eragiketak:
Eragiketa erlazionalak: <, <=, >, >=, /=, =



2.3. Adierazpenak

- Balio bat kalkulatzeko “formulak”
 - Aldagai bati balio berri bat esleitzeko aginduetan, adibidez
- Eragiketen arteko lehentasuna inplizituki definituta dago
- Eragile batzuek lehentasun handiagoa dute
Adibidez, $I+J/K$ adierazpena: $I+(J/K)?$ $(I+J)/K?$
- Lehentasun-maila bereko bi eragile batera erabiltzen direnean ezker aldetik elkartzen dira.
Adibidez, $I/J*K$ adierazpena: $(I/J)*K?$ $I/(J*K)?$
- Parentesiak beti erabil daitezke lehentasun esplizitua ezartzeko

Adierazpenak

Eragileak eta lehentasunak

Eragile diadikoak	Erag. monadikoak	Lehentasuna
<p>**</p> <p>* / mod rem</p> <p>+ - &</p> <p>= /= < <= >= ></p> <p>and or xor</p>	<p>abs not</p> <p>+ -</p>	<p>Handiena</p> <p>↑</p> <p>↓</p> <p>Txikiena</p>



Adierazpenak. Ariketa

Zein izango da adierazpen hauek ebaluatzeko ordena?

- Not Eguzkitsua or Euritsua
- $X > 4.0$ and $Y > 0.0$
- $-4.0 * A^{**2}$
- $\text{abs}(1 + A) + B$
- $A / B * C$
- $A / (B * C)$
- $(-4) * (A^{**}(5 + 1))$
- $(-4) * A^{**}(5 + 1)$
- $(A / (B * C))$
- $A * B / C$
- $\text{abs}(X - Y^{**2}) > 2.0 * X * 0.001$
- $(A / B) * C$
- $A + B * C$



2.4. Datuak manipulatzeko oinarrizko aginduak

- Datuen irakurketa
- Datuen idazketa
- Asignazioa

Helburua:

- Programetako objektuekin burutzen diren oinarrizko **ekintzak** ezagutzea eta erabiltzea.



Ekintza: Datu-irakurketa

- Datuak sekuentzia batetik irakurtzen dira.
- Sekuentziako elementu guztiak mota berekoak dira
- **Irakurri_Osokoa** (*aldagai1*)
sekuentzian oraindik irakurri gabe dagoen lehenengo osokoa *aldagai1* aldagaiari asignatzen dio



Ekintza: Datu-irakurketa

- Antzekoak

Irakurri_Erreala (*aldagai2*)

Irakurri_Karakterea (*aldagai3*)

- Sekuentziako elementuen mota eta aldagaiarena berdinak dira
- Sekuentziako elementu bat irakurtzeko aldezturik irakurri behar izan dira aurrean dauden guztiak.



Ekintza: Datu-idazketa

- Emaitzak beste sekuentzia batean idazten dira.

Idatzi_Osokoa (*adierazpen1*)

*adierazpena ebaluatuz lortzen den balioa
(osokoa) idatzi egiten du irteera-sekuentzian*

- Adibidez: Idatzi_Osokoa ($(X+Y)*2$)
- Antzekoak

Idatzi _Erreala (*adierazpen2*)

Idatzi_Karakterea (*adierazpen3*)

Idatzi_Katea (*adierazpen4*)



Ekintza: Asignazioa

<aldagaia> := <adierazpena>

- *Adibidez: $M := (N + P) * 6$*
- *$(N + P) * 6$ adierazpena ebaluatuz lortzen den balioa M aldagaiari ezarriko zaio balio berri bezala.*
- *M aldagaiak galtzen du lehenago zeukan balioa.*
- *Adierazpenean aldagairik azaltzen bada (N eta P), adierazpena ebaluatzen denean aldagaiak une horretan daukan balioa erabiltzen da.*
 - *Aldagaiaren balioa ez da aldatzen erabiltzeagatik*



Ekintza: Asignazioa (Adibideak)

Hasierako egoera:

N: 2 P:4 X : 1.0 Y : 4.5

Asignazioak

M := (N + P) * 6

Z := Y - X

P := P + 1

Asignazioak egikaritu ondoko egoera:

N: 2 **P:5** X : 1.0 Y : 4.5

M: 36 **Z : 3.5**



2.5. Kontrol-egiturak

- Algoritmoko ekintza-sekuentzia pausoz pauso eta ordenan egikaritzen da.
- Ordena sekuentzial lineal hori aldatzeko erabiltzen dira kontrol-egiturak:
 - Baldintzazko egiturak
 - Iterazioa



Baldintzazko egitura

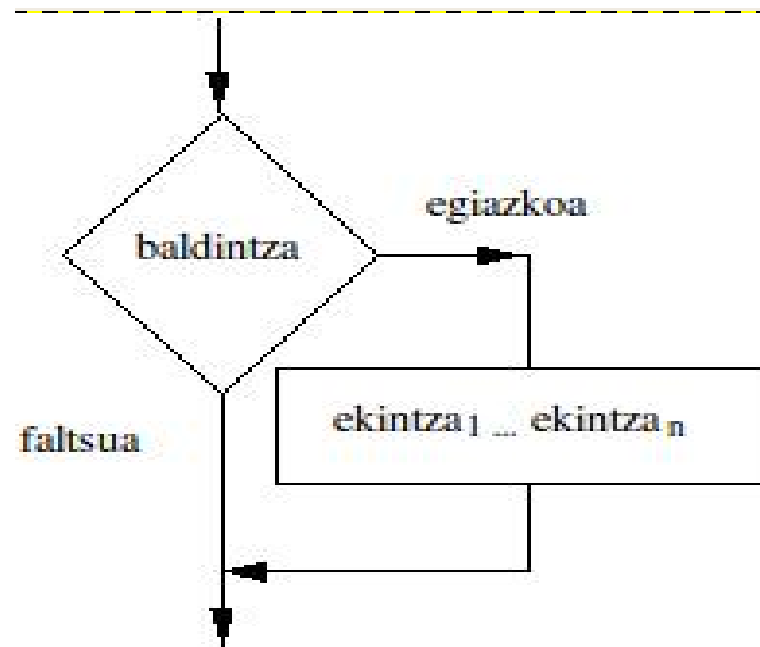
- Pauso bat (edo gehiago) **baldintza bat betez gero** egin behar bada

```
baldin baldintza orduan  
    ekintza1... ekintzan  
ambaldin
```

```
baldin baldintza orduan  
    ekintza1... ekintzan  
bestela ekintzab1... ekintzabn  
ambaldin
```

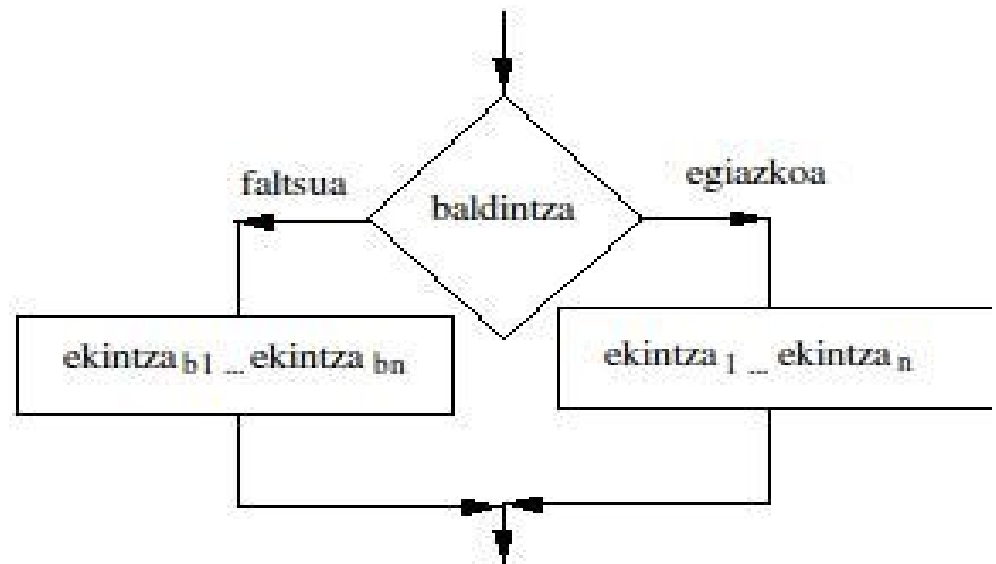
Baldintzazko egitura

baldin baldintza **orduan**
ekintza₁... ekintza_n
ambaldin



Baldintzazko egitura

baldin baldintza **orduan**
 $ekintza_1 \dots ekintza_n$
bestela
 $ekintza_{b1} \dots ekintza_{bn}$
ambaldin





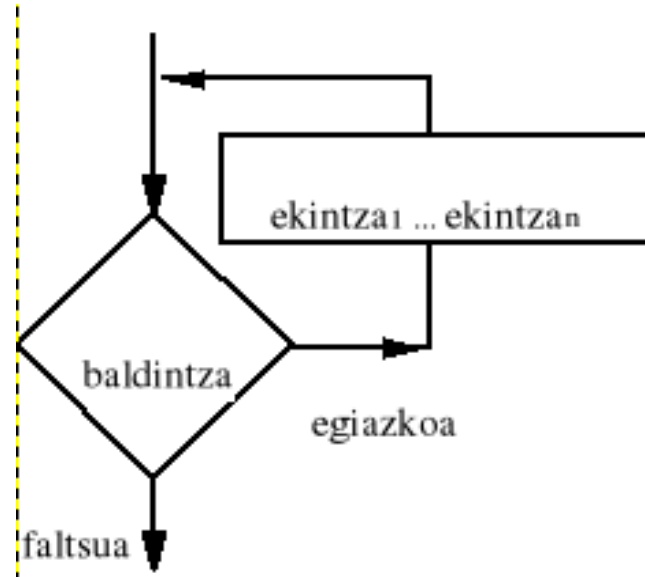
Iterazioa

- Pauso bat (edo gehiago) **errepikatu** behar direla adierazteko, **bitartean** eskema erabiltzen dugu:

```
bitartean baldintza egin  
    ekintza1  
    ekintza2  
    ...  
    ekintzan  
ambitartean
```


Iterazioa

bitartean baldintza **egin**
ekintza₁ ... ekintza_n
ambitartean

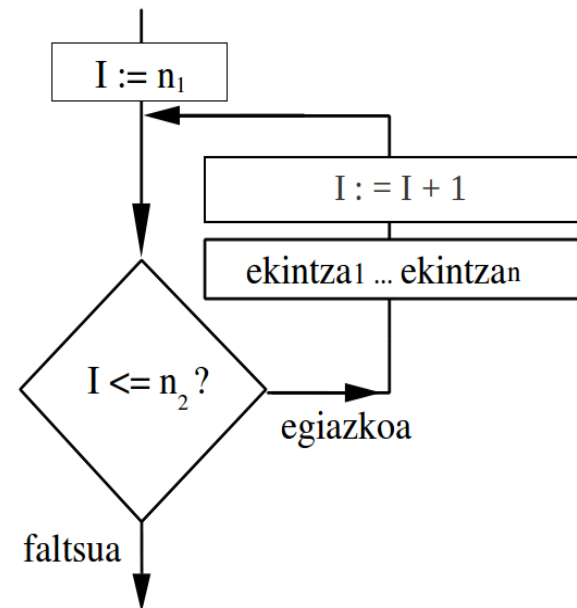


Aldi kopuru jakineko iterazioa

egin I guztietarako n_1 tik n_2 raino

$ekintza_1 \dots ekintza_n$

amguztietarako





guztietarako kontrol-egitura. Adibidea

Algoritmoa Kuboak_1_30

hasiera

egin I **guztietarako** 1 **tik** 30 **eraino**

Idatzi_Osokoa (I**3)

amguztietarako

amaia

Eraitza:

1 8 27 64 125 ... 27000



guztietarako kontrol-egitura

- Bi bertsio baliokide:

```
algoritmoa Kuboak_1_30
hasiera
    egin I guztietarako 1 tik 30eraino
        Idatzi_Osokoa (I**3)
    amguztietarako
amaia
```

```
algoritmoa Kuboak_1_30
hasiera
    I := 1
    bitartean I <=30 egin
        Idatzi_Osokoa (I**3)
        I := I + 1
    ambitartean
amaia
```



2.6. Algoritmoen idazkera

```
algoritmo <identifikadorea>  
hasiera  
    <agindua>*  
amaia
```



Algoritmoen idazkera. Adibidea

Irakurri triangelu baten oinarria eta altuera eta triangeluaren azalera idatzi.

algoritmo Kalkulatu_Azalera

hasiera

```
Idatzi_Katea (G,"Triangelu baten Azalera kalkulatu dugu")
```

```
Idatzi_Katea (G,"Zein da triangeluaren Oinarria?")
```

```
Irakurri_Erreala (F,Oinarria)
```

```
Idatzi_Katea (G,"Zein da triangeluaren altuera?")
```

```
Irakurri_Erreala (F,Altuera)
```

```
Azalera := Oinarria*Altuera/2.0
```

```
Idatzi_Katea (G,"Hau da triangeluaren azalera:")
```

```
Idatzi_Erreala (G,Azalera)
```

amaia

Algoritmoen idazkera (II)

Zehaztapenarekin eta aldagaien erazagupenarekin

algoritmo Kalkulatu_Azalera

```
-- Aurrebaldintza: F=<a,b>, non:  
    a,b: Errealak, eta a>0.0 , b>0.0  
  
-- Postbaldintza: G=<d>, non:  
    d: Erreala, eta d baita a oinarriko eta b altuerako triangeluaren azalera.
```

Zehaztapena

Oinarria, Altuera, Azalera: Erreal

Aldagaien erazagupena

hasiera

```
Idatzi_Katea (G,"Triangelu baten azalera kalkulatu dugu")  
Idatzi_Katea (G,"Zein da triangeluaren oinarria?")  
Irakurri_Erreala (F,Oinarria)  
Idatzi_Katea (G,"Zein da triangeluaren altuera?")  
Irakurri_Erreala (F,Altuera)  
Azalera := Oinarria*Altuera/2.0  
Idatzi_Katea (G,"Hau da triangeluaren azalera:")  
Idatzi_Erreala (F,Azalera)
```

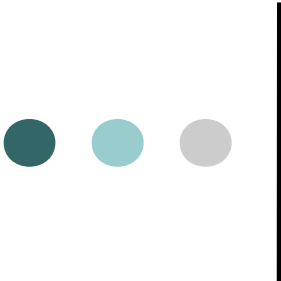
amaia

Aginduak



Oinarrizko eskemak

- Sekuentziako elementuen tratamendua
 - Korritze osoa
 - Sekuentziako elementu guztiak tratatzen dira.
 - Adibidez: kontatu zenbat 'a', bilatu maximoa.
 - Korritze partziala
 - Ez da ailegatzen sekuentziako elementu guztiak tratatzera.
 - Adibidez: Elementu bat bilatu.



Sekuentziako elementu guztien tratamendua (korritze osoa)

hasiera

Hasierakoak

Lortu lehenengo elementua

bitartean *elementua ez da azkena* **egin**

Tratatu elementua

Lortu hurrengo elementua

ambitartean

Tratatu azken elementua

Bukaerakoak

amaia



Karaktere-sekuentzia bateko elementu guztien tratamendua

hasiera

Hasierakoak

Irakurri_Karakterea (Kar)

bitartean *Kar* /= ' .' **egin**

Tratatu elementua

Irakurri_Karakterea (Kar)

amaitartean

Tratatu puntua

Bukaerakoak

amaia



Osoko-sekuentzia bateko elementu guztien tratamendua

- Suposatuz sekuentzia 0 zenbakiarekin bukatzen dela.

hasiera

Hasierakoak

Irakurri_Osoa (N)

bitartean N \neq 0 **egin**

Tratatu elementua

Irakurri_Osoa (N)

ambitartean

Tratatu zeroa

Bukaerakoak

amaia



Sekuentziako elementuen korritze partziala (elementu baten bilaketa)

hasiera

Hasierakoak

Lortu lehenengo elementua

Aurkitua := Faltsua

bitartean ez (*elementua azkena da*) **eta** Aurkitua=Faltsua **egin**

baldin *elementua bilatu bada orduan*

Aurkitua := Egiazkoa

ambaldin

Lortu hurrengo elementua

ambitartean

Tratatu azken elementua

Bukaerakoak

amaia



Karaktere-sekuentziako elementuen korritze partziala (karaktere baten bilaketa)

hasiera

Hasierakoak

Irakurri_Karakterea (Kar)

Aurkitua := Faltsua

bitartean Kar/='.' **eta** Aurkitua= Faltsua **egin**

baldin *elementua bilatu bada* **orduan**

Aurkitua := Egiazkoa

ambaldin

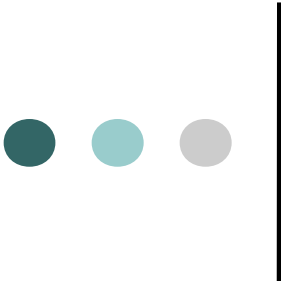
Irakurri_Karakterea (Kar)

ambitartean

Tratatu puntua

Bukaerakoak

amaia



Osoko-sekuentzia bateko elementuen korritze partziala (zenbaki bat bilatu)

hasiera

Hasierakoak

Irakurri_Osoa(N)

Aurkitua := Faltsua

bitartean N/=0 **eta** Aurkitua=Faltsua **egin**

baldin *elementua bilatu bada* **orduan**

Aurkitua := Egiazkoa

ambaldin

Irakurri_osoa(N)

ambitartean

Tratatu zeroa

Bukaerakoak

amaia