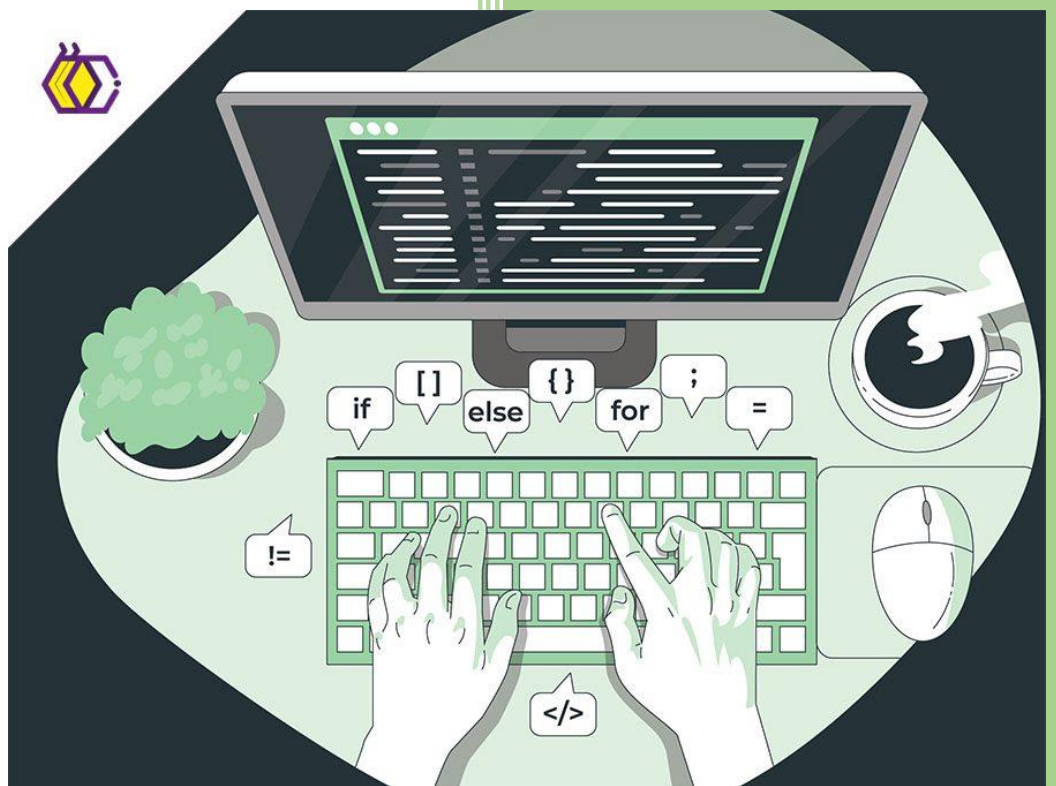


2024

# PROGRAMAZIOAREN PARADIGMAK



TALDE 5

AIA S.A

8-11-2024

## AURKIBIDEA

ZER DIRA? .....	2
PROGRAMAZIO PARADIGMAK IKASTEKO IMPORTANTZIA .....	2
PARADIGMA NAGUSIAK .....	3
PARADIGMA IMPERATIBOA .....	3
PROGRAMAZIO EGITURATUTA.....	4
PROGRAMAZIO PROZEDIMENTALA .....	5
PROGRAMAZIO OBJEKTUETARA ORIENTATUTA .....	5
PROGRAMAZIO KONKURRENTEA.....	7
PROGRAMAZIO PARALELOA.....	9
PARADIGMA DEKLARATIBOA .....	9
PROGRAMAZIO FUNTZIONALA .....	9
PROGRAMAZIO LOGIKOA.....	10
PROGRAMAZIO ERREAKTIBOA .....	10
PARADIGMA ERAGILETAN OINARRITUTA .....	11
PARADIGMA BANATUTA .....	12

## ZER DIRA?

Programazio-lengoiak ordenagailuei hainbat zeregin eta ekintza burutzen "irakasteko" erabiltzen dira. Lengoai bakoitzak bere hiztegia eta gramatika-arauak ditu, agindu horiek garatzeko.

Paradigma programazio-estiloa edo metodologia da, ez programazio-lengoaia bat. Paradigma batek arazoak kode jakin bat erabiliz ebazteko modua adierazten du. Programazio-lengoai ezagun asko daude, baina denek jarraitu behar dituzte hainbat arau ezartzean, eta arau horiek dira paradigmak, hau da, lengoiaren funtzionalitatearen arabera sailkatzeko bitartekoa.

Paradigma estilo, eredu edo programazio-metodologia bezala uler daiteke, eta helburua da arazoak lengoi jakin bat erabiliz modu eraginkorrean konpontzea.

Gainera, programazio-lengoaia berri bat garatzen denean, ohikoa da paradigma bakar batean edo batzuetan kokatzea, bere ezaugarrien arabera.

## PROGRAMAZIO PARADIGMAK IKASTEKO IMPORTANTZIA

Programazio-paradigmek kodea irakurgarriago eta antolatuago eraikitzen laguntzen dute. Gainera, aplikazio mota bakoitzerako teknikarik egokienak eskaintzen dituzte, garatzailearen eguneroko produktibitatea handituz. Lengoaiak sakonago ulertzeko eta kodeen artean azpiko esanahiak ulertzeko gaitasuna ematen dute.

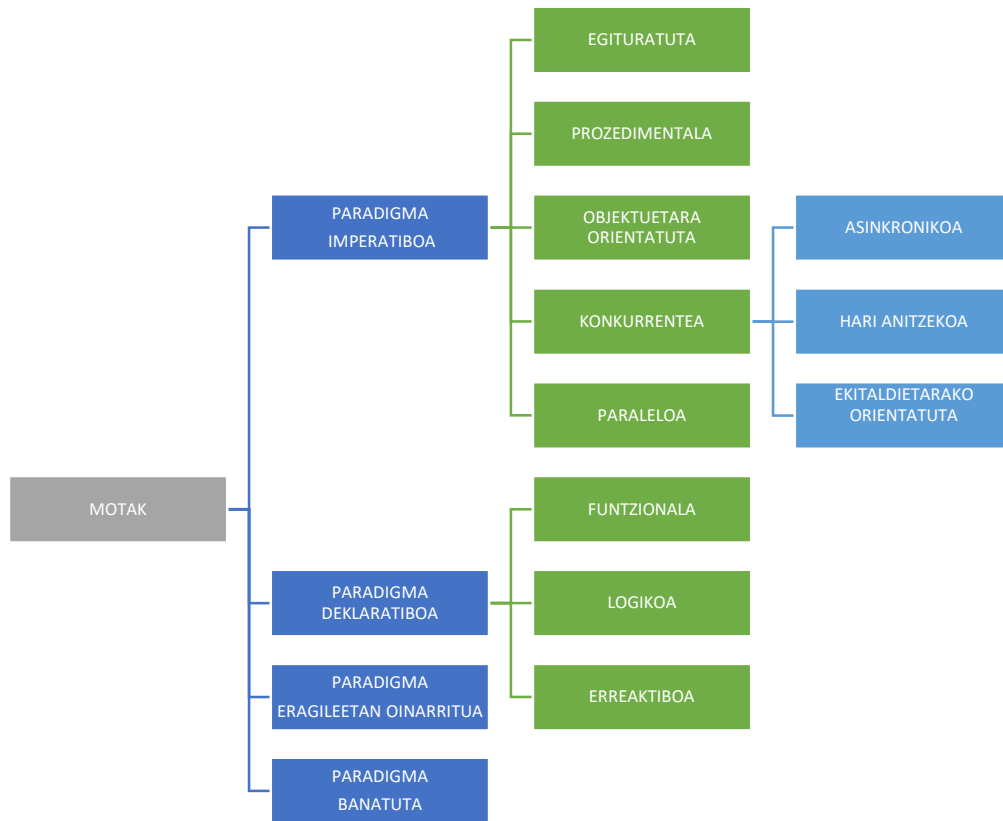
Programazioan, arazo bakoitzak hainbat irtenbide izan ditzake, horregatik esaten dugu arazoak konpontzeko paradigma bat baino gehiago dagoela. Beraz, sistema jakin baten garapenean, paradigma bat bestea baino abantailatsuagoa izan daiteke, aplikazio espezifikoko baterako teknikak egokiagoak eskainiz.

Proiektu baterako paradigma egokia hautatuz gero, aplikazioak modu produktiboagoan garatu daitezke, taldean kode idazteko orientazio bakarra ahalbidetuz, irakurgarriagoa eginez eta bere bizitza osoan zehar mantentzea erraztuz.

Programazio-paradigmak ulertzeak proiektua eta zuk zeu profesionalagoak eta antolatuagoak egingo zaituzte. Horrela, arazo baten irtenbidea pentsatu aurretik, irtenbide horren modelizazioa eta erabiliko den paradigma kontuan hartuko dira.

## PARADIGMA MOTAK

Hemen ikusten duzue dauden paradigma motak:



### PARADIGMA IMPERATIBOA

Programazio inperatiboa programazio-paradigma klasikoa da, eta bertan kodean urratsez urrats definitzen dira exekutatu beharreko aginduak. Paradigma honek programak modu sekuentzialean idazten ditu, hau da, aginduak ordenean eta jarraian exekutatzen dira, eta programatzaileak zehazten du nola lortu nahi den emaitza. Programazio inperatiboaren barruan hainbat **azpimota** daude:

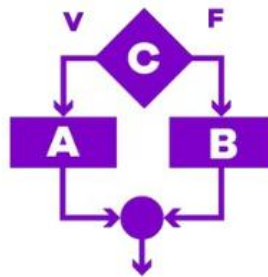
## PROGRAMAZIO EGITURATUTA

Programaren diseinua eta kodearen antolaketa hobetzeraz bideratuta dago. azpierrutinak edo funtzioak soilik erabiliz. Hiru kontrol-egitura nagusi erabiltzen ditu:

- **Sekuentzia:** Hizkuntzan modu naturalean ematen dena da, sententziak programan agertzen diren ordenan exekutatzen direlako, hau da, beste baten atzean.



- **Adarkatzea (if-else):** Baldintzapeko egituraren oinarria da sekuentzia bat aldagai boolear bati ematen zaion balioaren arabera exekutatzen dela.



- **Iterazioa (bucleak):** Errepikapen-egiturak sententzia bat edo multzo bat exekutatzen du, baldin eta aldagai boolearra egiazkoa bada.



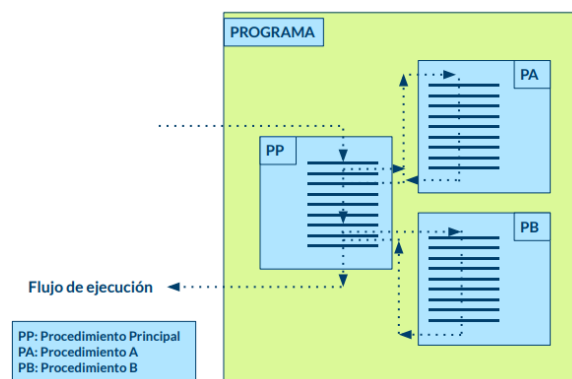
Paradigma egituratu mota hau programatzen irakasteko erabiltzen da. Jarraibideak goitik behera irakurtzen dira, eta ez da beharrezkoa kontzeptu edo gauza konplexuetan abstraitzea (ondo ondoko aginduak ematean soilik zentratzea).

## PROGRAMAZIO PROZEDIMENTALA

Prozedurazko edo prozedurazko programazioa programazio egituratutik dator. Kodea errutinak edo prozedurak izeneko atal logikoetan zatitzean datza; prozedura bakoitzak ataza jakin bat ebazten du, eta behar den bakoitzean exekutatzen da. Nabarmendu behar da prozedura batek beste prozedura batzuetara dei dezakeela.

Lehen prozedura-lengoietakoko batzuk:

- FORTRAN
- ALGOL
- COBOL
- BASIC



## PROGRAMAZIO OBJEKTUETARA ORIENTATUTA

Klaseen eta objektuen kontzeptuan oinarritzen da. Programazio mota hau software-programa bat pieza sinpleetan eta kode-planoen (motak) pieza berrerabilgarrietan egituratzeko erabiltzen da, objektuen banakako instantziak sortzeko.

Objektuetara Bideratutako Programazioaren paradigmarekin, programen logika hutsean zentratzeari utzi eta objektuetan pentsatzen hastea da bilatzen duguna, eta hori da paradigma honen oinarria. Horrek asko laguntzen digu sistema handietan; izan ere, funtzioetan pentsatu beharrean, sistemaren osagaien erlazioetan edo elkarrekintzetan pentsatzen dugu.

Klase batek definitzen du nola izango diren hainbat atributu dituzten objektuak, eta haien portaerak metodoak

funtsezko 4 pilaretan oinarritzen da:

- **Abstrakzioa:** Erabiltzaileak objektu baten ezaugarri eta metodo hautatuekin bakarrik elkarreragiten duenean, objektu konplexu batera sartzeko goi-mailako tresna sinplifikatuak erabiliz.
- **Enkapsulazioa:** Objektu baten barruan dagoen informazio garrantzitsu guztia biltzen du, eta hautatutako informazioa bakarrik azaltzen dio kanpoko munduari.
- **Herentzia:** Oinarrizko atributuen eta portaeren klase baten definizioaren bidez, bigarren mailako klaseak sor daitezke.
- **Polimorfismoa:** Mezu beraren aurrean instantzia desberdinetarako portaera desberdinak exekutatzea ahalbidetzen du

Paradigma hau erabiltzen duten programazio lengoai batzuk:

- Java.
- JavaScript.
- Python.
- C++
- PHP.

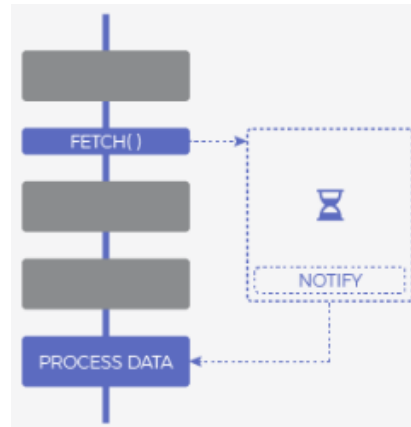






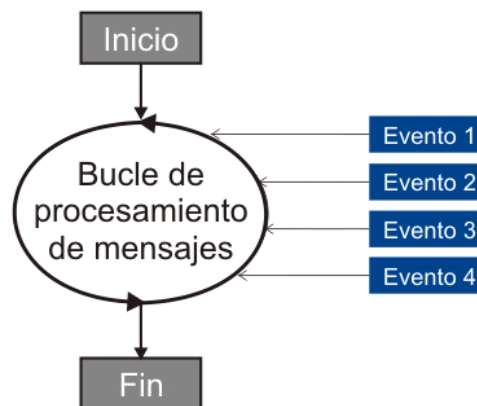
- PROGRAMAZIO ASINKRONIKOA

Hari nagusia blokeatzen ez duten atazak gauzatzean oinarritzen da, haiek osatu zain dauden bitartean. Eragiketak bigarren planoan egiten dira, hari nagusiak beste zeregin batzuk egiten jarrai dezan.



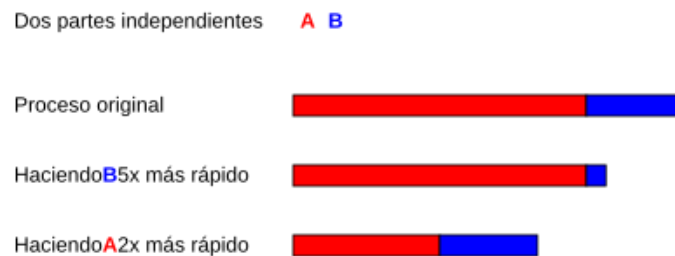
- EKITALDIETARAKO ORIENTATUTA

Gertaerei erantzuteko kodearen exekuzioan oinarritua. Ekitaldien begizta bat erabiltzen da gertaeren etorrera maneiatzeko eta zereginen gauzatzea koordinatzeko, ekitaldi horien arabera.



## PROGRAMAZIO PARALELOA

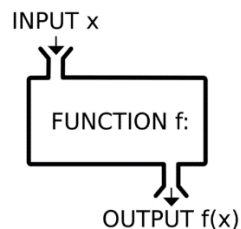
Jarduera bera aldi berean zenbatzeko hainbat baliabidetan prozesatzeko gaitasuna da, eta, horrela, exekuzio-denbora murrizten da.



## PARADIGMA DEKLARATIBOA

### PROGRAMAZIO FUNTZIONALA

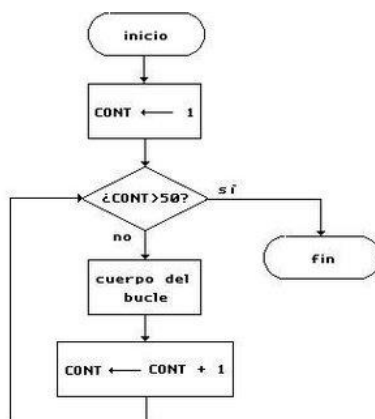
Programazio-problema ebazteko funtzio matematiko hutsak erabiltzera bideratzen da. Programa baten egoera eta aldaketak azpimarratu beharrean, datuen kalkuluan eta transformazioan zentratzen da. Funtzio puruek emaitza bera itzultzen dute beti input beragatik, eta ez dute bigarren mailako ondorioz sistema. Hori dela eta, programak aurreikusteko modukoak eta probatzeko errazagoak dira.



## PROGRAMAZIO LOGIKOA

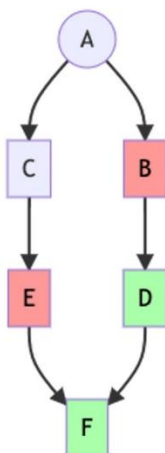
Erabiltzaileak ordenagailua programatzeaz arduratu behar duen mekanismoa da, logika matematikoaren postulatueta eta parametroetan oinarrituta bere erabakiak hartzeko gai izan dadin.

Programazio mota horrek bere helburua bete ahal izateko, programatzaileak anbiguotasunik gabeko kode bat idatzi behar du, eta horrek esan nahi du idatzizko sententziek zentzua izan behar dutela eta egiazkoak izan behar dutela. Horrek esan nahi du logikoak direla.



## PROGRAMAZIO ERREAKTIBOA

Datu-fluxuei eta gertaerei buruzko operadore gisa datuak maneiatzean oinarritzen da, egoera-aldagai estatikoen ordeiz. Behagarriak eta Behatzaileak funtsezko elementuak dira ikuspegi horretan, datu-fluxu horien aldaketen aurrean kudeatzeko eta erreakzionatzeko aukera ematen baitute.



## PARADIGMA ERAGILETAN OINARRITUTA

Programazio-paradigma bat da, eta "eragilez" osatutako sistemak sortzen ditu. Sistema horiek erakunde autonomoak dira, eta ingurunea hautemateko, arrazoitzeko eta horren arabera jokatzeko gai dira. Eragile horiek softwarea eta hardwarea izan daitezke, eta zeregin espezifikoak egiteko edo arazo konplexuak modu independentean konpontzeko diseinatuta daude.

Aplikazioak:

- **Adimen artifizialeko sistemak:** Laguntzaile birtual eta robot gisa.
- **Jolasak:** Pertsonaiek (NPCek) modu autonomoan joka dezakete.
- **Simulazioak:** Sistema konplexuetan portaerak modelatzeko, hala nola ekosistemetan edo merkatuetan.

Agenteei zuzendutako programazioari esker, ohiko ikuspegiekin konpontzen zailak diren arazoei aurre egin dakieke, eragileen autonomia eta egokitze gaitasuna aprobetxatuz.



## PARADIGMA BANATUTA

Osagaiak kokapen desberdinetan dituzten sistemen diseinuan eta inplementazioan oinarritzen den paradigma da, baina elkarrekin lan egiten dute helburu komun bat betetzeko. Osagai horiek gailu, sare edo leku geografiko desberdinetan egon daitezke, eta sareen bidez komunikatzen dira elkarren artean.

Aplikazioak:

- **Datu-base banatuen sistemak:** Datuak hainbat zerbitzaritan biltegitzen dira.
- **Hodeiko zerbitzuak:** Internet bidez baliabideak eta aplikazioak erabiltzeko aukera ematen dute.
- **Peer-to-peer sareak:** Erabiltzaileek beren artean baliabideak zuzenean parteka ditzaten.

Mikroszerbitzu-aplikazioak: aplikazio bat zerbitzu txiki eta autonomo askotan banatzen dutenak.

Programazio banatua funtsezkoa da erabilgarritasun, eskalagarritasun eta sendotasun handia eskatzen duten sistema modernoak sortzeko.

