

1. GAIA

INFORMATIKAREN OINARRIZKO KONTZEPTUAK

1 DEFINIZIO OROKORRAK

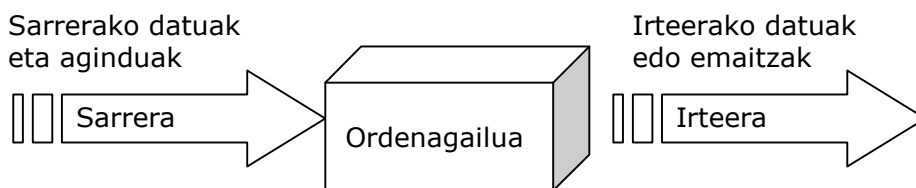
Ikus ditzagun definizio batzuk:

Informatika:

- Informazioaren tratamendu automatikoa arduratzen den zientzia.
- Ezaguera zientifiko eta teknikoaren multzoa, ordenagailu bitarteko informazioaren tratamendu automatikoa ahalbidetzen duena. (RAE)
- Gizarte, ekonomi eta teknika arloan ezaguera eta komunikazioen euskarritzat hartutako informazioaren tratamendu automatiko eta arrazionalaren arduraduna den zientzia.
- Ezaguera arloa, ordenagailuen diseinu eta erabilerarako alderdi guztiak biltzen dituena.

Ordenagailua:

- Makina elektroniko digitala da, emaitza lortu arte sarrerako datuekin aritmetika-logiko eragiketak burutzen dituena.



- Ordenagailua, hortaz, emaitza edo irteera bat lortzeko informazioa prozesatzen duen makina bat da. Programa baten egikaritzea tratamendu batzuen burutzea ekartzen du ondorioz, sailkatutako agindu multzo baten (hau da, programa baten) arabera. Ordenagailu batek programa bat egikaritu dezan, mi motatako informazioa ematea beharrezkoa da:
 - Programaosatzen duten instrukzioak (aginduak).
 - Programaren eragiketarako behar diren datuak.

Hardware eta software:

- Ordenagailu baten euskarri fisikoa edo hardwarea makina bera da, hau da, zirkuitu elektroniko, kable, karkasa, gailu elektromekaniko eta bera konposatzen duten beste osagaien multzoa.
- Ordenagailu baten zati materiala konposatzen duten osagaiek. (RAE)
- Euskarri logikoa edo softwarea ordenagailuan dauden programa exekutagarrien multzoa (sistema eragilea, erabiltzaileen aplikazioak).
- Zenbait ataza ordenagailu batetan egikaritzeko programa, agindu eta arau informatikoen multzoa (RAE).

Memoria:

- Ordenagailu baten memoria hardwarearen zatia bat da, denbora tarte baten barruan datuentzako informazioa, emaitza partzial, sistemak behar dituen programak osatzen dituzten aginduak atxikitzen duena, modu egokian funtzionatzeko.
- Geroago erabiltzeko datu eta aginduak gordetzen dituen gailu fisikoa, normalean elektronikoa (RAE).

Etxe-ordenagailu batetan, normalean memoria itxura hauek hartzen ditu:

- Disko gogorra: erabiltzailearen fitxategi eta programak gordetzen ditu. Informazioa kontserbatzen du, ordenagailua itzaltzen bada ere, eta edukiera handiko memoria da.

Adibidea:

Disko gogorrean gure programak gordeko ditugu, hurrengo saioan haietara sarbidea izateko, berreditatzeko, eta abar.

- RAM Memoria: ordenagailua erabiltzen den bitartean, aldi baterako programak eta informazioa edo horrelako gauzak gordetzeko erabiltzen da. Hau egiten da mota honetako memoria azkarra delako eta baliagarria ere datuetara sarbidea izateko, beste euskarri geldoago batetatik irakurri behar ez izateko, adibidez, disko gogor batetatik. Informazioa galtzen da ordenagailua itzaltzen denean.
- Cache Memoria: normalean mikroprozesadore, disko gogor eta beste gailu batzuetan aurkitzen da. Memoria kantitate txikia da, baina abiadura handikoa, eta ordenagailuko beste atal batzuk abiadura maximoan lan egin dezaten erabiltzen da, sistemako beste atal motelago batzuei behin eta berriro informazioa eskatu beharrean.
- Memoria sekundarioa: Datuen bilketa masiborako gailu periferikoen multzoa da, memoria nagusia baino edukiera handiagokoa, baina azken hau baino geldoagoa. Disketea, disko gogorra edo finkoa, unitate optikoak, flash memoriak edo ZIP diskoak maila honetan sartzen dira. Gailu periferiko hauek memoria nagusi edo barneko memoriara estekatuta geratzen dira, ordenagailuaren memoria subsistema osatuz. Memoria sekundarioko euskarriak: CD, CD-R, CD-RW DVD, DVD-/R, DVD-/RW, disketea, disko gogorra, zinta magnetikoa, pendrivea.

Adibidea:

Memoria sekundarioan gure datuak gordeko ditugu beste ekipo batetara eraman nahi baditugu, beste saio batetan eta beste ekipo batetan beraietara sarbidea edukitzeko, berreditatzeko, eta abar.

Periferikoak:

- Ordenagailua zentzu gabeko makina izango litzateke kanpoaldearekin komunikatuko ez balitz, hau da, periferikorik ez balu. Periferiko lege izendatzen dira, bai kanpoaldearekin komunikatzeko ordenagailuak erabiltzen dituen gailuak, bai memoria nagusiko osagarri bihurtuz, informazioa gordetzen duten gailuak, karkasa kanpoan ala barruan badago ere.
- Hauen artean bereiztu dezakegu:
 - o Sarrerako unitateak, exekutatu nahi diren programak eta beharrezko datuak sartzeko (teklatura, arratoia).
 - o Irteerako unitateak, ordenagailuak programen emaitzak eman ahal izateko (pantaila, inprimagailua)
 - o Memoria masiboa edo osagarria, funtzionamendua edo erabilera errazteko (pendrive).

Agindua:

- Agindu bat ordenagailuarentzat ariketa edo tratamendu bat adierazten duen sinbolo multzo bat da.
- Letra eta zenbakiz osatutako adierazpena, ordenagailu batetan burutu behar den eragiketa eta beharrezko datuak erakusten duena (RAE).

Adibidea:

```
printf ("Kaixo");
```

Programa:

- Programa bat ordenagailuari ematen zaion ordenatutako agindu multzoa da.
- Bateratutako agindu multzoa, ordenagailu bateri hainbat funtzio burutzea ahalbidetzen duena, adibidez, testu tratamendua, grafikoen diseinua, ariketa matematikoen ebazpena , datu-bankuen erabilera, eta abar. (RAE)

Adibidea:

```
#include <stdlib.h>
#include <stdio.h>
void main()
{
    char a,b;
    scanf("%c,%c",&a,&b);
    printf ("%d",a);
    printf ("%d",b);
    system("PAUSE");
}
```

Memoria, sarrera, irteera, agindua:



Despacho del Sr. Controlador



El Sr. Controlador ordena los datos que le llegan por la ventanilla de entrada y los coloca en el archivador para utilizarlas



El Sr. Controlador ejecutando una orden



El Sr. Controlador envía los resultados a través de la ventanilla de salida

Programazio-lengoaia:

Aginduak arau zehatzak jarraituz eta errepertorio jakin bateko sinboloekin osatzen da. Programazio-lengoaia bat artifiziala da eta programa baten eraikuntzan erabiltzen diren eta ordenagailu batekin komunikazioa ahalbidetzen duten arau, sinbolo eta hitz bereiziaz konposatzen da.

Maila altuko lengoaia:

Lengoaia naturalekoak bezalako zeinu konbentzionalen bitartez ordenagailu batekin komunikatzea errazten duen lengoaia. Abstrakzio maila altu dago esketzen dena eta ordenagailuak benetan ulertzen duenaren artean. Baita, erlazio konplexua dago maila altuko lengoaia eta makina codearen artean. Normalean, maila altuko lengoiak ikasteko errazak dira, lengoaia naturaleko elementuez osatuta daudelako, adibidez, ingelesa.

Adibidez:

```
if (zenbatzailea == 0)
    printf ("Balioa zero da");
```

Programazio-lengoaia batetan kontzeptu bitaz hitz egin dezakegu: **sintaxia** edo sententziak idazteko dagoen arau multzoa (arauak jarraitzen ez dituen sententzia OKERRA izango da) eta **semantika** edo sententzien esangura (semantika okerreko sententzia, arau sintaktikoak betetzen ditu, baina ez du egiten guk nahi duguna sententzia egokia erabili ez dugulako).

Makina lengoaia:

Zuzenean interpretatu eta exekutatu daitezkeen kodifikatutako agindu multzoa. Lengoaia hauek, ordenagailu batek ulertu ditzaken 0 eta 1-eko konbinazio ezberdinez osatuta daude. Programa exekutagarriak makina lengoia idatzita daude.

Zergatik bakarrik 0 eta 1? Ordenagailua zirkuitu digitalen multzo bat besterik ez delako, eta zirkuituek ulertzen duten gauza bakarra, sistema bitar bat da (hau da, 1=korrontea, 0=korronte eza).

Fitxategi exekutagarria:

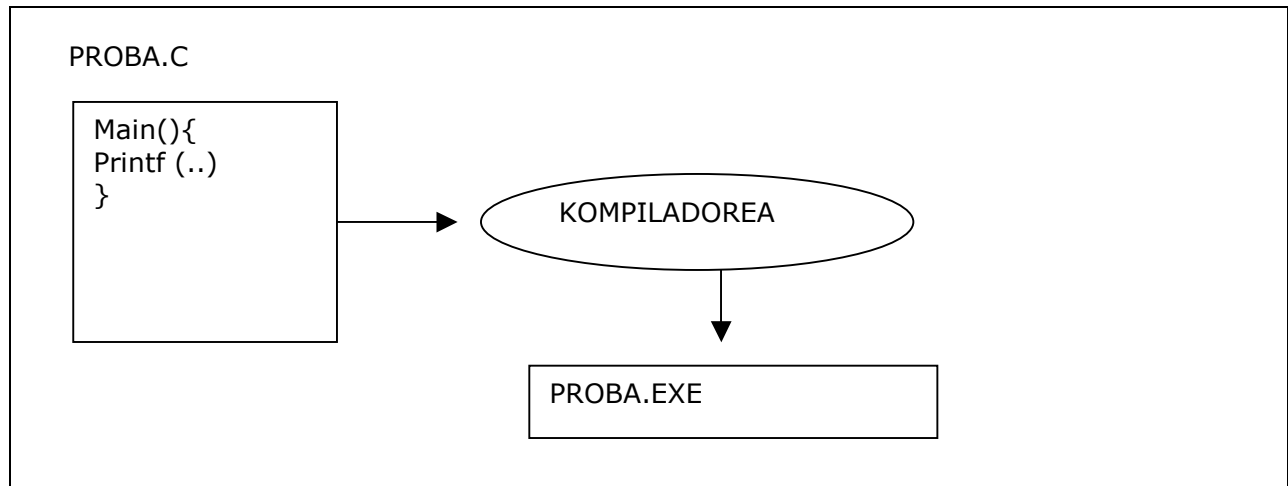
Exekutagarri bat, edo fitxategi exekutagarri bat, informatikan, ordenagailu batek programa bat bezala hartzen duen fitxategi bat da. Normalean prozesadore jakin baterako makina kodean dauden aginduak edukitzen ditu, baina exekutatzeke interpretatu behar duen bytecode bat ere eduki dezake. Gainera, sistema eragile jakin baterako sistemarako deiak zehatzak ere edukitzen ditu. Agindu motaren arabera, exekutagarri mugikor (plataforma ezberdinetan exekutatu daitekeena) edo ez-mugikorrei (plataforma bakarrarentzat) buruz hitz egingo dugu. Java adibidez, mugikorra da, prozesadore jakin batera lotuta ez doan bytecode bat erabiltzen duelako.

Kompiladorea:

Erabiltzailearen lengoaia informatikoa ordenagailuaren berezko lengoiara itzultzen duen programa.

Adibidea:

C programazio lengoia idatzitako programazioa kode exekutagarrian bihurtzen duen aplikazioa da. Guk lehenago agertzen den bezalako C lengoia idatzitako programa editatzen dugu eta zuzenean exekutatu dezakegun programa eskuratzen dugu.

**Sistema eragilea:**

Sistema informatiko baten oinarritzko prozesuen kudeaketa eta beste eragiketen egikaritze normala burutzen duen programa edo programa multzoa. Bere funtzio nagusiak exekuzio denboraren banaketa, sarrera/irteeraren kudeaketa, memoria kudeaketa eta informazio kudeaketa dira. Sistema eragile on baten ezaugarriak eraginkortasuna, fidagarritasuna, mantentze-lan erraztasuna eta baliabide gutxien erabilera dira.

Adibideak:

MS-DOS, Windows, UNIX, LINUX, OS/2, OS/400, AIX

Aplikazioak:

Erabilera jakin baterako prestatutako programak, esaterako, testu prozesadoreak, kalkulu orriak, datu baseak, aplikazio grafikoak, soldata ordainketak, eta abar.



Algoritmo:

- Pausu sekuentzia mugatua, ordenatua eta ez anbigua, problema baten ebazpena deskribatzen duena.
- Problema baten ebazpena lortzea ahalbidetzen duen eragiketen multzo ordenatu eta mugatua (RAE).
- Algoritmo baten ezaugarriak:
 - Zehatza.
 - Pausuen ordena adierazi behar du.
 - Egoera berdinetan beti jokatzeko modu berean.
 - Amaiera izan behar du.
 - Ez dago programazio-lengoiara lotuta.

Adibide ez-informatico bat:

6 pertsonentzat patata tortilla egitea

OSAGIAK:	TRESNAK:
6 arrautza fresko	platera, zartagina
1 Kg patata	aiztoa, sardexka

PRESTAKUNTZA

1. Patatak zuritu eta zatitu, platerean utziz
2. Bol batetan, arrautzak irabiatu
3. Patatak arrautza irabiatuekin nahastu
4. Zartagina olio apur batekin berotu
5. Arrautzak patatekin zartaginera bota...

Pertsona bakoitzak gauza bera adieraziko luke pausu ezberdinekin edo modu ezberdinean (beste hitz batzuekin, ...). Horregatik, algoritmo bat deskribatzeko **pseudokodea** erabiliko dugu, burutzeko pausu eta eragiketak azaltzen dituen zeinu konbentzionaleko sistema. Pseudokode hau "Notazio algoritmikoaren gida azkarra" eranskinean azaltzen da.

2 NOLA EGINGO DUGU GUK LAN?

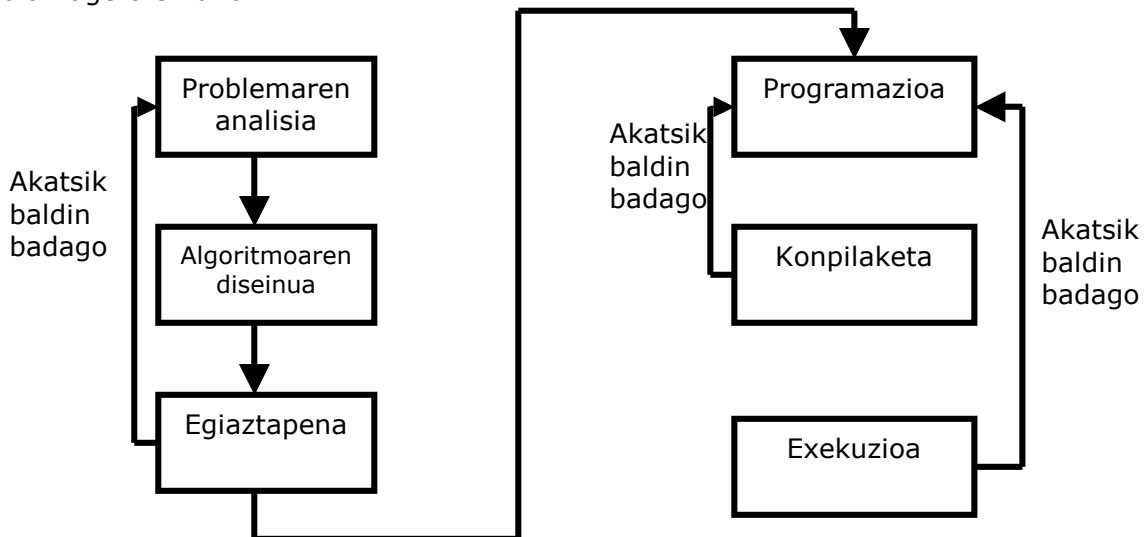
Laburpen bat egiten, C lengoaia erabiliz guk ebazteko ariketa bat eskuratuko dugu. Horretarako:

1. Ebazpen algoritmoa pentsatu eta pseudokodea idatziko dugu.
2. Ebazpen hori maila altuko programazio-lengoaia batera itzuliko dugu, C lengoaia (datuak eta aginduak). C lengoiaren arauak jakin behar izango ditugu, zerbait ingelesez azaltzeko hitzak eta arauak jakin behar ditugun bezala, eta ezin izango ditugu alde batera utzi, ordenagailuak gu ulertzea nahi badugu.
3. Programa C programazio-lengoaia idatziko dugu, konpilatzerako ere balio izango zaigun programa berberaz, hau da, gure programa makina lengoiara itzuliko du (0 eta 1) eta exekutagarri bat sortuko du, ordenagailuak aginduak egikaritu ahal izateko.



Prozesu honetan atzera bueltatzera behartzen gaituen akatsak egon daitezke, proposatutako algoritmoa egokia ez dela ikusten badugu, edo konpilaketa sintaxi akatsak ematen badizkigu (C lengoian idazten dugunean, konpilatzaileak "ez duela ulertzen" esaten digu arauak jarraitu ez ditugulako)edo baita exekuzioan ikusten dugu ebazpenean erratu garela, programak guk nahi duguna egiten ez duelako kasu guztietan (izan leike guk idatzitakoaren esangura edo semantika okerra izatea).

Prozesu hau, zehaztasun handiagoz eta egiaztapenak eta atzera itzuliak adieraziz, hurrengo irudian agertzen dira:



Guzti honetarako konpilatzailea lortzea ezinbestekoa izango da. Gure kasuan **DevC ++** izango da, eta gure ordenadorean instalatu behar izango dugu. Hemendik aurrera C ikasteko egokieran egongo gara gure lehenengo programa idatzi ahal izateko.

