PSEUDOKODEA

Pseudokodea kodea programazio-lengoaia batetik hurbil dagoen lengoaia bat da, eta programatzaile batek erraz interpreta ditzakeen algoritmoak garatzea du helburu, programazio-lengoaia edozein dela ere. Berez, ez da programazio-lengoaia bat, baina adierazpen-multzo mugatu bat erabiltzen du, programazio egituratu eta modularreko paradigmetan deskribatutako kontrol-egiturak eta moduluak ordezkatzeko.

Pseudokodearen bidez, soluzio finitua izan eta abiapuntu bakar batetik hasten diren algoritmoak idatz daitezke. Algoritmo edo programa bat pseudokodean idazteak, ondoren, aukeratutako programazio-lengoaiara itzultzea erraztu beharko luke.

| **Ba al zenekien…**  pseudokodean idazteko sintaxi estandarizaturik ez dagoenez, desberdintasun nabarmenak aurki daitezke programatzaile desberdinek idatzitako pseudokodeetan.. |
| --- |

# Eragileak, hitz erreserbatuak eta datu motak

Nahiz eta ez dagoen arau zurrunik programak sasikodean nola idatzi behar diren ezartzen duenik, gomendagarria da programa erraztasunez erabiliko den programazio-lengoaiara transkribatzea ahalbidetuko duten zenbait gomendio jarraitzea. Jarraian, operadoreen laburpen-taulak, hitz erreserbatuak eta sasikodea idazteko erabilitako datu-motak agertzen dira.

**Operadoreen laburpena:**

| **Aritmetikoak** | | **Erlazionalak**  **(Baldintzak sortzeko erabiliak)** | | **Logikoak**  **(Baldintzak sortzeko erabiliak)** | |
| --- | --- | --- | --- | --- | --- |
| + | Gehiketa |
| - | Kenketa | = | Berdin | and | eta logikoa |
| \* | Biderketa | < | Txikiago | or | edo logikoa |
| / | Zatiketa |  | Txikiago edo berdin | no | ez logikoa |
| div | Zatiketa osoa | > | Handiago | **Bereziak** | |
| mod | Ondarra |  | Handiago edo berdin | ← | esleipena |
| ^ | Potentzia | <> | Desberdin | // | iruzkina |

**Hitz erreserbatuak:**

| Inicio | Si no | Otro | Para | En |
| --- | --- | --- | --- | --- |
| Fin | Según | Mientras | Hasta | Procedimiento |
| Si | Hacer | Repetir | Incremento | Función |
| Entonces | Caso | Hasta que | Cada | Imprimir |
| Leer | Retornar |  |  |  |
| Carácter | Cadena | Entero | Real | Booleano |

# Kontrol-egiturak

Aurreko atalean aipatu den bezala, pseudokodeak programazio egituratuaren berezko kontrol-egiturak erabiltzen ditu. Horregatik, kontrol sekuentzial, alternatibo eta iteratiboko egiturak irudikatzen dituzten sekuentziak erabiliko dira.

## Kontrol sekuentzialeko egiturak

Azalpenaren hurrenkeran (sekuentzialki) exekutatzen diren jarraibideen blokeak deskribatzen ditu. Multzoak HASIERA-AMAIERA adierazpenek muga ditzakete edo beste egitura batzuetan egon daitezke. Jarraian, egitura sekuentzialaren adibide bat agertzen da.

| Inicio  <instrucción1>  …  <instrucciónN>  Fin |
| --- |

## Kontrol alternatiboko egiturak

Kontrol alternatiboaren edo selektiboaren egiturak jarraibideen multzo baterantz edo besterantz bideratzen du gauzatze-fluxua, baldintza jakin baten gainean egiten den ebaluazioaren arabera.

Kontrol-egitura mota horren azpimota desberdinak daude:

1. Alternatiba sinplea: balio boolearra itzuliko duen baldintza bat betetzen bada exekutatuko diren jarraibideen multzoa ezartzen du.

| Si <condición> entonces  <instrucción1>  …  <instrucciónN>  Fin Si |
| --- |

1. Alternatiba bikoitza: baldintza betetzen ez bada exekutatuko den beste argibide-bloke bat.

| Si <condición> entonces  <instrucción1>  …  <instrucciónN>  Si no  <instrucción1>  …  <instrucciónN>  Fin Si |
| --- |

1. Aukera anitza: hainbat argibide-bloke exekutatzeko aukera ematen du, kasu bakoitzaren balioekin alderatzen den adierazpen batek hartzen duen balioaren arabera edo baldintza baztertzaileak dituzten alternatiba bikoitzeko egiturak lotuz. Emaitzako bloke baliokideen adibidea:

| Según <expresión> hacer  Caso <valor1>  <instrucción1>  Caso <valor2>  <instrucciones2>  Caso <valor3>  <instrucciones3>  Otro caso  <instruccionesN>  Fin según | = | Si <condición1> entonces  <instrucciones1>  Si no  Si <condición2> entonces  <instrucciones2>  Si no  Si <condición3> entonces  <instrucciones3>  Si no  <instruccionesN>  Fin Si  Fin Si  Fin Si |
| --- | --- | --- |

## Kontrol iteratiboko egiturak

Kontrol iteratiboaren egiturak aukera ematen du jarraibideen bloke bat baldintza bat betetzen den bitartean exekutatzeko.

Kontrol-egitura mota horren hainbat azpimota daude:

1. Hasieran irteera duen Iterazioa (while): lehenik, baldintza ebaluatzen du eta, betetzen bada, jarraibideen blokea exekutatzen du. Baldintzak balioa aldatu beharko du, emandako jarraibideen arabera, begizta amaigabeak saihesteko. Litekeena da jarraibideen blokea inoiz ez exekutatzea.

| Mientras <condición> Hacer  <instrucciones>  Fin Mientras |
| --- |

1. Amaieran irteera duen iterazioa (repeat eta do while): lehenik, jarraibideen blokea exekutatzen du eta, ondoren, baldintza ebaluatzen du. Baldintzak balioa aldatu beharko du, begizta amaigabeak saihesteko jasotako jarraibideen arabera. Jarraibideen blokea, gutxienez, aldi batez exekutatuko da. Repeat erabiltzen duen bertsioak jarraibideak betearaziko ditu baldintza betetzen den arte, eta direktoa, berriz, while aldaera exekutatzen den bitartean.

| Repetir  <instrucciones>  Hasta Que <condicion> |  | Hacer  <instrucciones>  Mientras <condicion> |
| --- | --- | --- |

1. Iterazioa kontagailuarekin (for): jarraibideen blokea iterazio-kopuru jakin batez exekutatuko du. Iterazio bakoitzeko gehituz edo txikiagotuz joango den aldagai bat erabiltzen du, irteera-baldintza bete arte. Kontrol-baldintza (i = = N edo i = = 1, gehikuntzaren arabera) faltsua izango da jarraibideak betetzen jarrai dezaten. For inkremental eta dekrementalaren adibideak.

| entero i  Para i ← 1 Hasta N Incremento 1 Hacer  <instrucciones> |
| --- |
|  |
| entero i  Para i← N Hasta 1 Incremento -1 Hacer  <instrucciones> |

1. (For each) bakoitzerako iterazioa: adibide multzo batean jasotako elementu bakoitzerako jarraibideen blokea exekutatuko du

| entero i  Para Cada elemento En conjunto Hacer  <instrucciones>  Fin Para Cada |
| --- |

## Egitura modularrak

Era berean, sasikode bidez, programen deskonposaketa modularra deskriba daiteke, zereginak egiteko prozedurak eta funtzioak adierazita.

### Prozedurak

Prozedura bateko kode-bloke bakoitzak argibide-multzo bat eramango du burura, hala eskatzen zaionean. Dagokion parametroa definitu bada, deitzeko orduan lan egiteko moduko argudioak jaso ahal izango ditu, baina inola ere ez da irteera-baliorik itzuliko. jarraian, prozedura baterako pseudokodea adibide bat agertuko da:

| Procedimiento nombre (tipo parametro1, …, tipo parametroN)  <instrucciones>  Fin Procedimiento |
| --- |

Ondoren dago beste adibide bat “Kaixo Mundua” eta “Agurra”:

| Procedimiento HolaMundo ()  imprimir (“Hola Mundo”)  imprimir (“---------”)  Fin Procedimiento  Inicio  HolaMundo()  Fin |  | Procedimiento Saludo (cadena nombre)  imprimir (“Hola + nombre)  Fin Procedimiento  Inicio  cadena nombre  imprimir (“Introduzca su nombre: “);  leer (nombre)  Saludo(nombre)  Fin |
| --- | --- | --- |

### Aldagaiak

Aldagai bat datu batzuk gordetzeko ordenagailuko memorian dagoen leku baterako izena da.

| Definir vble1,...,vbleN datuMonta (Real, Entero, Logico, Caracter) |
| --- |

### Funtzioak

Prozedurek bezala, funtzio batek ere lan egiteko argudioak jaso ahal izango ditu deian, dagokion parametroa definitu bada, baina, prozeduretatik urrun, irteera-balio bat itzuliko da. Funtzio baterako pseudokodean adibidea:

| Función nombre (tipo parámetro1, tipo parámetro2, …): tipoRetorno  <instrucciones>  …  retornar X  Fin Función |
| --- |

Ondoren, funtzioa erabiltzen duen programa baten adibidea:

| Función TextoImpar (entero número) : cadena  cadena resultado  Si (número % 2 =0 ) entonces  resultado ← “par”  Si no  resultado ← “impar”  Fin Si  Fin Función  Inicio  entero número  imprimir (“Introduzca el número …”)  leer (numero)  imprimir (“el número introducido es “ + TextoParImpar (número))  Fin |
| --- |

# Fluxu diagramak

Fluxu-diagrama bat algoritmo edo prozesu baten irudikapen grafiko bat da. Informatikan erabiltzen dira, baina beste eremu batzuetan ere erabiltzen dira. Algoritmoa ulertzea errazten du, algoritmoa gauzatzeko fluxuari buruz ematen duen deskribapen bisualari esker.

## Sinbologia

Prozesuari esanahia emateko, elkarren artean lotuz doazen sinbolo-multzo bat erabiliz eraikitzen dira fluxu-diagramak. 1.4 koadroan erabilitako sinbolo bakoitza agertzen da, baita hauen deskribapen bat ere:

* fluxu-diagrama obalo edo elipse baten bidez irudikatutako terminal batekin hasten eta amaitzen da.
* Trapezio angeluzuzenak datuak teklatutik sartzea adierazten du. Prozesuaren geldialdia adierazten du, erabiltzaileak datuak tekleatu arte.
* Datuen sarrera edo irteerarako periferikoekiko komunikazioa paralelogramo baten bidez adierazten da.
* Algoritmo guztietan ordena bat dago, eta ordena horretan egiten dira eragiketak. Fluxu-diagrametan ordena hori gezi baten bidez adierazten da.
* Erabakiak erronbo baten bidez adierazten dira, eta erronbo horretatik ahal beste fluxu-lerro ateratzen dira.
* Adierazi beharreko algoritmoa oso handia bada, konplexua izan daiteke diagrama bloke bakar batean diseinatzea, eta egokiena diagrama bloke txikiagoetan banatzea izango da. Horretarako, konektoreak erabil daitezke, puntu horretara iritsitakoan jauzi egiteko erabiliko den puntua irudikatzeko. Jauziak zirkunferentzia baten bidez irudikatuko dira orri berean badira, edo pentagono irregular baten bidez beste batean badira.
* Barneko kalkulu-eragiketak egiten dituzten prozesuak laukizuzen baten bidez irudikatzen dira.
* Litekeena da beren konplexutasunagatik prozesu gisa irudikatu ezin diren zeregin jakin batzuk egotea. Kasu horretan, beste leku batean definituta egongo diren eta alde bakoitzean lerro bikoitza duen laukizuzen batez irudikatuta dauden azpiprozesuetan deskonposatuko dira.
* Informazioa trukatzeko erabiltzen diren datu-baseek beren ikurra dute (zilindroa).
* Inprimatutako dokumentuek ere beren irudikapen-sinboloak dituzte.

Fluxu diagramen sinbologia:

|  |
| --- |
|  |

## Kontrol egiturak

| Sekuentziala | Aukera Simplea | Aukera bikoitza |
| --- | --- | --- |
| Aukera anitzak IF baldintza habiaratuekin | | |
| Aukera anitzak | | |
| **ERREPIKAPENAK** | | |
| While | Repeat | do while |
| For iterativa dekrementala    FOR iterativa inkrementala | | |

**FUNTZIOAK**

Funtzio edo prozedura bakoitza (egitura modularrak) diagrama independente baten bidez deskribatuko da, eta azpiprograma horretarako deia diagrama nagusitik egingo da.

* Prozedura bat deitzeko, azpiprozesura egindako dei baten bidez irudikatuko dugu. Azpiprozesu hori alde bikoitzen laukizuzenak identifikatzen du. jarraian, pseudokodean egindako adibideen diagramak erakusten dira.

