

## C PROGRAMAZIO-LENGOAIA (XIII)

### C konpiladorea eta beste lengoaiak

Iñaki Alegria & Montse Maritxalar

HAMAIKAGARREN kapituluaren estekatzailak (linker-ak) aparte konpilatutako modulu desberdinak programa exekutagarri bakar batean lotu ditzakeela ikusi genuen. Beraz lengoia desberdinetako moduluak aparte konpilatu ondoren, liburutegian biltegiturik ala ez, objektu-moduluak direnez estekatzailaren bidez bil daitezke.

Dena den, PCko mihiztadura-lengoiaz idatzitako errutinak C programa batetik deitzeko zenbait puntu hartu behar da kontutan eta hori izango da kapitulu honetako lehen pasarte. Bestetik, PC bezalako 16 biteko ordenadoreetan C programak exekutatzeko konpilatzean aukera desberdinak egin behar dira, bigarren atalean ikusiko dugun legez. Azkenik, hirugarren pasartean, sistema eragilearen jatorrizko funtzioak, sistema-deiak deitutakoak, nola erreferentzia daitezkeen aztertuko dugu.

#### C eta mihiztadura-lengoia

PCetan exekutatzen diren programak memoriako 6 eredu desberdinetan antola daitezke konpiladorearen arabera. Hurrengo pasartean aztertuko ditugun eredu hauen arabera, erakusleak ondoko hiru mota honetakoak izan daitezke: near, far eta huge.

near motako erakusleak 16 bitekoak diren bitartean, gainerakoak 32 bitekoak dira; lehen kasuan helbidea segmentu berean dagoela suposatzen bait da. far eta huge motako erakusleak antzekoak dira (biak 32 bitetan gordetzen bait dira) eta haien artean dagoen desberdintasun bakarra desplazamenduan adieraz daitezkeen neurrian datza.

near, far eta huge hitz gakoak dira. PCetako C konpiladoreetarako eta erakusle, array zein funtzioen definizioetan erabil daitezke. C programa batek mihiztadura-lengoiaz idatzitako errutina bat erabiltzeko, ez du ezer berezirik egin behar; funtzio bezala erazagutu eta erreferentziatzea besterik ez bait du behar. Dena den, funtzioaren erazagupenean near, far edota huge erabil daiteke memoriako ereduaren arabera.

Aldiz Ctik deitutako errutina bat mihiztadura-lengoiaz idazten badugu, ondoko arauak hartu behar dira kontutan:

parametroak pila kokatuko dira, jeneralean aldeantzik ordenean, hau da, azken parametrotik hasiko da pila betetzen. Parametroek hartuko duten luzera 2. kapituluaren aztertutako datuen ezaugarrien arabera izango da.

errutinan, dagozkion ekintzak baino lehen, honako beste hauek burutu egin behar ditu: 1) erabiliko diren erregistroen edukina (BP beti, eta SI, DI, SS, DS edota CS beharrezkoa bada) pila gordetzea. 2) BP erregistroan SPren balioa gordeko da; horrela BPren bidez parametroak eskuratuko bait dira. Lehen parametroa (BP + 4) helbidean egongo da funtzioa near bada eta (BP + 6)-an gainerako kasuetan. Aldagai lokalak erabiltzeko SP erregistroaren balioa txikiagotuz joango da.

errutinari dagozkion ekintzak burutu ondoren, itzul-emaizta AX erregistroan kokatu behar da eta ondoren gordetako erregistroen balioa berreskuratu.

1. programan bi zenbaki biderkatzen dituen mihiztadura-lengoiaz idatzitako bider izeneko errutina azaltzen da, segmentuei dagozkien sasiaginduak azaldu gabe. Aipatzekoa da mihiztadurazko errutinen izenari azpimarra karakterea aurretik jartzen zaiola.

	PUBLIC	_bider
_bider	PROC	NEAR
	push	bp
	mov	bp, sp
	mov	ax, [bp + 4]
	cmul	WORD PTR [bp + 6]
	mov	sp, bp
	pop	bp
	ret	
bider	ENDP	

1. programa. Bider errutina mihiztadura-lengoiaz.

## C konpiladorea 16 biteko ordenadoreetan

Esan dugunez PCetako programetan memoriako 6 eredu desberdinen artean aukera egin daiteke konpilatzeko garaian. PCetako memoria-sistema segmentatua da eta bertan lau segmentu bereizten dira (aginduetakoa, datuetakoa, pilakoa eta estra), bakoitzari erregistro bat dagokiolarik; CS, DS, SS eta ES hain zuzen. Erregistro hauen balioa (eta ondorioz, programaren luzera) aukeratzeko ereduak daude.

Small ereduaz bi segmentu baino ez dira erabiltzen; aginduetarako eta datuetarako. Beraz, programek gehienez 128 K edukiko dute. Programa txikietarako erabiltzen da. Medium ereduaz datuetarako segmentu bat eta aginduetarako segmentu anitz adierazten da. Beraz, datuek 64 K baino gehiago hartzen ez duten bitartean, aginduek Mega bateraino irits daitezke. Datu gutxi maneiatzen duten programa luzeetarako da. Compact ereduaz, aurrekoaren kontrakoa egiten da: aginduek 64 K baino gutxiago hartu eta datuek Mega bateraino. Datu asko maneiatzen duten programetarako egokia da. Large ereduaz, agindu zein datuak Mega bateraino irits daitezke, bietan segmentu anitz egokitzen delako. Beraz, programa handietarako egokia da. Huge ereduaz, Large ereduaren antzekoa da, baina datu estatikoak segmentu bakarrean kokatzen dira.

Eredua handiagotu ahala, programa luzeagoa eta motelagoa gertatzen da, erregistro eta pilaren arteko trukeak direla eta.

Seigarren eredu Tiny izenekoa da eta Turbo C konpiladoreak baino ez du eskaintzen. Eredu honen arabera programa osoa segmentu bakarra da eta 64 K baino gehiago ezin du hartu.

Eredua konpilatzeko komandoan zehazten da, eta Microsoft-eko C konpiladorea honela erabiltzen da:

```
/AS small ereduaz
/AM medium "
/AC compact "
/AL harge "
```

## Sistema-deiak C-tik

Sistema-deiak sistema eragilearen jatorrizko eragiketak dira. Beraz, sistema eragilearen arabera sistema-dei desberdinen aurrean egongo gara.

Hemen azalduko ditugun sistema-deiak bi sistema eragile garrantzitsuenak dira: MS-DOS eta UNIX-enak.

### MS-DOS

dos.h fitxategian sistema-deiak erabiltzen dituzten liburutegiko funtzioak erazagutzen dira. Honako hauek dira esanguratsuenak:

<i>funzioa</i>	<i>esanahia</i>
absread	irakurketa
abswrite	idazketa

disable	etenak galeraztea
getdfree	diskoko erabili gabeko tokia
inport	portu batetik irakurtzea
int86	eten bat sortzea
keep	bukatu eta egoliar mantentzea
outpor	portu batean idaztea
setveet	eten-bektorea aldatzea
bdos	sistema-deia
intdos	sistema-deia

bdos eta intdos funtzioetan parametroaren arabera dei desberdinak egin daitezke.

### UNIX

UNIXen sistema-dei asko daude, sistema eragilearen zerbitzuak zuzenean erabiltzeko asmoz. Izenburu-fitxategi desberdinetan dago sistema-deiei dagozkien funtzioen erazagupena; definizioa liburutegian konpilaturik bait dago.

Honako hauek dira dei garrantzitsuenak:

#### prozesuen gaineko deiak

exec	programa berri baten karga
fork	prozesu berri baten sorrera
wait	sortutako prozesu baten bukaera arte gelditzea
exit	programaren bukaera
sleep	zenbait segundo gelditzea

#### fitxategi eta gailuen gaineko deiak

open, read, write, close eta lseek aurreko kapitulu bezala

creat	fitxategi bat sortzea
link	fitxategi bat konpartitzea
mknod	azpikatalogo bat sortzea
chwod	babes-kodea aldatzea
utime	denbora-irakurketa
stat, fstat	fitxategiaren egoera
coctl	gailuen ezaugarri-aldaketa

#### prozesu arteko komunikazioa

pipe	pipe (komunikazio-bide sekuentzial) bat sortzea
dup	fitxategiaren deskribatzailea bikoiztea
semget	semaforo bat eskuratzea (System
semop	semaforoaren gaineko eragiketa (System

Sistema-dei hauen erabilpena nahiko konplexua eta bertsioaren araberakoa da. Horrexegatik ez dugu hemen gehiago sakonduko.

Hurrengo bi kapituluetan, azken bietan hain zuzen, aurrekoetan zehar aztertutako sententziak eta funtzioak erabili egingo ditugu, adibide konplexuak programatuz. E