

Konpilazioa I

<http://ji.ehu.es/konpi1>

Irakasleak: Eneko Agirre, Nerea Ezeiza, Julian Gutierrez
Telefonoak, e-mail helbideak, tutoretzak eta materiala web-gunean

Helburuak:

Irakasgaiaren helburua bikoitza da. Alde batetik nahi da ikaslea goi-mailako programazio-lengoaien eraikitzaileak objektu-kodera itzultzeko erabilitako teknikak ezagutzea, eta bestetik ikaslea gai izatea ezagutza hori eremu desberdinetako problema praktikoetan aplikatzea.

Filosofia orokor honen barruan, ondoko aspektuak azpimarratuko dira:

- Arloko kontzeptu nagusiak ezagutzea: konpilazioa, interpretazioa, itzulpen gurutzatua, bootstrapping, linkedition,...
- Konpiladorearen fase desberdinak eta bakoitzean aplikatutako teknika nagusiak ezagutzea.
- Konpiladoreen eraikuntzarako laguntza-tresnak ezagutzea eta erabili ahal izatea, bereziki UNIX/Linux inguruneetan (LEX eta YACC).
- Ikaslea ikasitako ezagutza teorikoa erabiltzeko gai izatea itzulpen arlo zabaleko problemak ebazteko, klasean aztertutako itzultzaileen eraikuntzarako metodoen bidez.

Praktika

Ikasturtean zehar honako praktika eraman beharko da aurrera:

Itzultzaile sinple baten diseinu eta inplementazioa, lehen hiru gaietan garatutako ideietan oinarrituaz (10 aste)

Praktika hau urritik abendura (urtarrilera) garatuko da, eta bere helburua ikasleak itzultzaileak eraikitzeko oinarritzko teknikak ezagutzea eta erabiltzea da. Praktikaren ebaluazioa lortutako helburuen arabera egingo da.

Eskola praktikoak

Eskola praktikoak teorian azaldutakoa osatzeko moduan antolatu dira, planifikazio paraleloarekin, baina antolaketa eta metodologia bereizi batekin. Eskola praktikoen helburua alde batetik lan jarraitua eta taldeko lana sustatzea da, eta bestetik ebaluazio osoago baterako datuak lortzea eta ikasleei irakasgaia ebaluatuko den erari buruzko pista konkretuak ematea.

Eskola praktikoetara bi modutara joan daiteke: *libreki* (ez du eraginik ebaluazioan) edo *araututa* (elementu osagarria izango da ebaluaziorako). Araututako eran praktika egingo duten ikasleak bakarrik eman dezakete izena, eta ondorengo ezaugarriak dauzka: asistentziaren kontrola, taldeko lana eta entrega eta derrigorrezko aurkezpenen araubidea.

Ebaluazioa

Irakasgaiaren ebaluazioa praktika eta bukaerako azterketaren kalifikazioan* oinarrituko da. Irakasgaiaren azken nota **azterketa** eta **praktika osoaren** arteko batezbestekoa izango da. Irakasgaia gainditzeko era edozein izanda ere, bai oinarrizko praktika baita oinarrizko azterketa ere biak gainditu behar dira.

Azterketara aurkezteak deialdia kontsumituko du. Praktika soilik aurkezteak ez.

Aurreko ikasturteko praktika gainditu zutenek ez dute oinarrizko praktika (5) berriz egin behar, kontrakoa ez badiote adierazten irakasleari (adibidez nota igotzeko).

(*) Eskola praktikoa arautuan parte hartu eta **ebaluazio positiboa** jasotzen dutenek azterketako nota puntu baten igoko dute 4tik aurrerako kalifikazioa badute.

Irakasgaiaren gaiak

1. gaia Sarrera.
2. gaia Analisi lexikoa.
3. gaia Itzultzaile sinple baten definizioa eta garapena.
4. gaia Analisi sintaktikoa.
5. gaia Exekuzio-denborako memoria kudeaketa.

Bibliografia:

- Aho, A.V.; Lam, M.S.; Sethi, R.; Ullman, J.D.: "*Compilers. Principles, Techniques and Tools. Second Edition.*". Ed. Addison-Wesley Computing. 2006.
- Aho, A.V.; Sethi, R.; Ullman, J.D.: "*Compilers. Principles, Techniques and Tools*". Ed. Addison-Wesley. 1986.
- Aho, A.V.; Sethi, R.; Ullman, J.D.: "*Compiladores. Principios, Técnicas y Herramientas*". Ed. Addison-Wesley Iberoamericana. 1990.
- Loudon, K.C.: "*Construcción de Compiladores. Principios y prácticas*". Ed. Thomson. 2004.
- Jim Holmes. Object-Oriented Compiler Construction. Prentice-Hall -1995
- Hunter, R.; "Compilers. Their Design and Construction Using Pascal" John Wiley & Sons Ltd. 1985.
- Miguel, J. y Blanco, J.M. LEX: Un generador de reconocedores léxicos. (Master en Ciencia y Tecnología de los Computadores - 1993)
- Miguel, J. y Blanco, J.M. YACC: Una herramienta para el desarrollo de traductores (Master en Ciencia y Tecnología de los Computadores - 1993)
- S. Pemberton, M.C. Daniels Ellis Pascal Implementation (2 volúmenes), Horwood Limited - 1983
- Schreider, A.T.; Friedman, H.G.Jr.; "Introduction to Compiler Construction with UNIX". Ed. Prentice-Hall: Englewood Cliffs, NJ. 1985.
- Teufel, Schmidt & Teufel Compiladores. Conceptos fundamentales. Addison-Wesley Iberoamericana – 1995
- Waite, T.; Goos, G.: "Compiler Construction". Ed. Springer-Verlag. 1984.
- D. Watson High-Level Languages and their Compilers, Addison-Wesley, 1989

1. Gaia: Sarrera

- 1.1 Konpiladore eta itzultzaileak
 - Interpretatzaileak
- 1.2 Programazio-lengoaien ezaugarriak
 - Eboluzio historikoa
 - Ezaugarri nagusiak
 - Programazio-lengoaien definizioa
 - Lengoaiaren egitura, datu-motak eta kontrol-egiturak
 - Programazio-lengoaien sailkapena
- 1.3 Konpiladore baten egitura
 - analisi lexikoa
 - analisi sintaktikoa
 - analisi semantikoa
 - erroreen tratamendua
 - sinbolo-taularen kudeaketa
 - bitarteko kodearen sorkuntza
 - kodearen optimizazioa
 - kodearen sorkuntza
 - front-end / back-end
- 1.4 Pasaldi anitzeko konpiladoreak
- 1.5 Konpiladorearen software-ingurunea
 - Aurreprozesadoreak
 - Mihiztatzaileak
 - Linkeditore eta kargatzaileak
 - Dekonpiladoreak
- 1.6 Konpiladoreen eraikuntzarako tresnak
 - Analizatzaile eta itzultzaileen sorkuntza
 - Kodearen sortzaile automatikoak
- 1.7 Konpiladoreen espezifikazio eta diseinua
 - T-diagramak
 - Bootstrapping (abioa)
 - Itzulpen inkremental/elkarreragilea
 - UNCOL proiektutik JVM-era
 - Konpilazio gurutzatua

Oinarrizko erreferentzia: 1. eta 11. kapituluak [Aho, 90] 1. kapitulua [Aho, 06; Louden, 04]

Erreferentzia osagarriak:

[Weiser, 92], 1, 2 eta 3 kapituluak [Watson, 89], 1 eta 2 kapituluak [Waite, 84], 2 eta 17 kapituluak [Sánchez, 84], eta 1., 11 eta 12 kapituluak [Sanchís eta Galan, 86].

Helburuak:

Gai honek oinarrizko helburu bat dauka, ikasleak konpiladore baten egitura orokorra ezagutu dezala, egitura horren arabera azalduko dira eta irakasgai honen edukiak. Gainera programazio-lengoaiei itzulpenean erabiltzen diren kontzeptu, termino eta tresna jakin batzuek ezagutzea era lortu nahi da, hala nola:

- Konpiladore eta interpretatzaileen arteko ezberdintasuna, bakoitzaren egitura ezagutuaz.
- Konpilazio teknika eta programazio-lengoaiei garapenak, eta beraiei arteko harremana.
- *Aurreprozesadore, pasaldi anitzeko konpiladore, dekonpilazio* bezalako terminoak.
- Itzultzaileak sortzen dituzten tresnak ezagutzea.
- Konpiladoreak espezifikatzeko T-diagramak erabiltzen jakitea.
- *Bootstrapping, itzulpen elkarreragilea eta konpilazio gurutzatua* ezagutzea.

2. gaia. Analisi lexikoa

2.1 Sarrera

- Tokenak eta atributuak
- Bere erlazioa konpiladorearen beste faseekin
- Iturburu-lengoiaren ezaugarrien arabera zenbait problema
- Errore lexikoak

2.2 Oinarrizko kontzeptuak

- Lengoaia erregularrak
- Adierazpen erregularrak eta definizio erregularrak
- Automata finituak

2.3 Analizatzaile lexiko baten diseinu eta inplementazioa

- Analizatzaile lexiko baten eraikuntza automata finitu determinista batetik abiatuta
- Inplementazioari buruzko zenbait kontsiderazio:
 - Hitz erreserbatuen tratamendua
 - Buffer-en erabilera
 - Bukaerako egoerekin lotutako ekintzen inplementazioa
 - Trantsizio funtzioaren inplementazioa

2.4 Lex: analizatzaile lexikoen espezifikaziorako lengoaia

Oinarrizko erreferentzia: 3. kapituluak [Aho, 90; Aho, 06], 2. kapituluak [Louden, 04] eta [Miguel, 93a].

Erlazioak beste gaiekin: 1. gaia ikusi ondoren, gai hau autonomoa da besteengandik.

Helburuak: Kapitulu honen helburu nagusia da ikaslea gai izatea analizatzaile lexiko eraginkorrak eraikitzeke. Honen bukaeran ikasleak gai izan behar du:

- Analizatzaile lexiko baten espezifikazioa egiteko adierazpen erregularrak erabiliz.
- Sarrerako katearen gaineko automata finitu determinista baten interpretazioan oinarritutako analizatzaile lexiko bat diseinatu eta inplementatzeko.
- Analizatzaile lexikoen eraikuntzan sortzen diren problema praktikoak menperatzeko: buffer-en erabilera, hitz erreserbatuen ezagutza edo errore lexikoen tratamendua.
- Analizatzaile lexiko bat egiteko LEX tresna erabiliz.

3. gaia. Itzultzaile sinple baten definizioa eta garapena

- 3.1 Sarrera
- 3.2 Lengoaiaren definizio sintaktikoa
 - Testuingururik gabeko gramatikak
 - Analisi zuhaitzak
 - Anbiguitasuna
- 3.3 Sintaxiak zuzendutako itzulpena
 - Sintaxiak zuzendutako itzulpen eskema
 - Analisi zuhaitzaren sakonerako korritzea
 - Atributu sintetizatuak
- 3.4 Analisi sintaktikoa
 - Goitik beherako analisia
 - Analisi auresalea
 - Analizatzaile auresaleen eraikuntza
 - Gramatika auresalea lortzeko eraldaketa
- 3.5 Programazio-lengoaien zenbait eraikitzailearen itzulpena
 - Adierazpen aritmetikoak
 - Adierazpen boolearrak
 - Baldintzazko sententziak
 - Agindu errepikakorrak
- 3.6 Analisi semantikoa
 - Egiaztapen estatiko eta dinamikoa
 - Moten egiaztatzaile baten espezifikazioa
 - Moten baliokidetasun eta bihurteta
 - Funtzio eta eragile "gainkargatuak"
- 3.7 Sinboloen taula
 - Sinboloen taularen edukia eta maneia
 - Sinboloen taularen errepresentazioa
 - Informazioaren irismena
- 3.8 Bitarteko kodearen motak
 - Hiru helbidetako kodea
 - Pila bidezko makina abstraktuak
 - o »JVM
 - o »p-code
 - Notazio postfixua, zuhaitz sintaktikoak eta ASTak
- 3.9 Programazio-lengoaien zenbait eraikitzailearen itzulpen hedatua
 - Erazagupenak
 - Esleipen-aginduak
 - Adierazpen boolearrak
 - Taulen erreferentziak

Oinarrizko erreferentzia: 2., 4., 5., 6. eta 8. kapituluak [Aho, 90], 2., 4., 5. eta 6. kapituluak [Aho, 06], 3., 4. eta 6. kapituluak [Louden, 04].

Erreferentzia osagarria: 2. kapitula [Fischer, 88].

Erlazioak beste gaiekin: 1. gaiarekin, gehienbat konpiladorearen faseen aurkezpenarekin. 2. gaian aurkeztutako analizatzailearekin. Aldi berean, 4. eta 5. gaitan aurkeztuko diren edukien sarrera da.

Helburuak: Gai honen helburu nagusia da ikasleak goi mailako programazio-lengoaia baten itzultzaile baten eraikuntzaren problema bideratu ahal dezan, era sistematikoan eta irakasgaiaren lehen pausoak ematen ari denean. Gai hau bukatutakoan ikasleak gai izan behar du:

- Ezker errekurtsibitatea edo aurrizki komunak dituen gramatika batetik (ezaugarri horiek ezinezkoa egiten baitute goitik beherako itzulpen auresalea) baliokidea den baina problema horiek ez dituen gramatika lortzea.
- Gramatika batetik abiatuta goitik beherako analizatzaile auresale errekurtsiboa lortzeko, eta bere funtzionamendua ulertzeko.
- Konplexutasun erdiko itzulpena espezifikatzea sintaxiak zuzendutako itzulpen-eskemak erabiliz, eta espezifikazio horiek inplementatuko dituen programa bat eraikitzeko.
- Analizatzaile lexikoa berari tokenak bata bestearen atzetik eskatzen dizkion analizatzaile sintaktiko batekin erlazionatzeko.
- Identifikatzeko konpilazio-denborako eta exekuzio-denborako murrizpenen egiaztapenak.
- Itzultzaile konkretu baten beharrentzako egokia den sinboloen taula diseinatzeko, eta testuinguruaren murrizpenen arabera errepresentazio desberdinen artean aukeratzeko gai izateko.
- Bitarteko kodeen mota desberdinak ezagutzeko.
- Sintaxiak zuzendutako itzulpen-eskema baten bidez programazio-lengoiaren eraikitzaile arrunten itzulpena espezifikatu ahal izateko, besteak beste adierazpen boolearrak, baldintzazko eta agindu errepikakorrak, sekuentziazioaren etenak edo taulen erreferentziak.

Edukiak: Gai hau *front-end*-en eraikuntzarekin lotuta dauden teknika konkretu batzuen sarrera da: goitik beherako analizatzaile errekurtsibo auresaleak eta sarrera-lengoiaren sintaxian oinarritutako espezifikazioak. Gaia paraleloan problema “sinple” baten soluzioarekin garatzen da: itzultzaile baten diseinua, zeinek sarrera bezala adierazpen aritmetikoa hartuz, adierazpen horren itzulpena hiru helbideko kodean sortuko duen. Honela, ondoko kontzeptuak lantzen dira: sintaxian oinarritutako espezifikazioa hasieran, ondoren analizatzaileak eraikitzeko metodoa, itzulpenaren espezifikazioa eta analisi-fasea erlazionatzen dira, lengoiaren espezifikazio lexiko eta sintaktikoa bereizten dira, eta itzultzaile osoaren diseinuarekin bukatuko da. Eduki hauek jaso eta gero ikasleek has dezakete tamaina bateko lengoaia baten itzultzailearen diseinua.

4. gaia. Analisi sintaktikoa

- 4.1 Analizatzaile sintaktikoa.
 - Bere erlazioa konpiladorearen beste zatiekin.
 - Goitik beherako eta behetik gorako analisia.
- 4.2 Iturburu-lengoiaren definizio sintaktikoa.
 - Testuingururik gabeko gramatikak.
 - Anbiguotasuna.
- 4.3 Goitik beherako analisia.
 - Analizatzaile auresalea: LL(1) baldintzak.
 - Goitik beherako analisi errekurtsiboa.
 - Goitik beherako analisi ez errekurtsiboa.
- 4.4 Behetik gorako analisia.
 - Laburtze-desplazamenduaren bidezko analisia.
 - LR analizatzaileak.
- 4.5 Errore sintaktikoen tratamendua.
- 4.6 Yacc: analizatzaile sintaktikoen sortzailea.

Oinarrizko erreferentzia: 3. eta 4. kapituluak [Aho, 90; Aho, 06] , 3., 4. eta 5. kapituluak [Louden, 04] eta [Miguel, 93b].

Erlazioak beste gaiekin: 3. gaian aurkeztu diren analisi sintaktikoari lotutako teknikak era metodikoago eta formalago baten aztertzen dira.

Helburuak:

Gai honen helburu orokorra ikaslea analizatzaile sintaktiko eraginkorrak diseinatu eta inplementatzeko gai izatea da. Zehatzago, ikasleak:

- Analisi auresalearen oinarriak ezagutzea
- Testuingururik gabeko gramatikak erraztasunez erabili ditzala, bai lengoiak espezifikatzeko baita analizatzaile sintaktikoak eraikitzeke ere. Beraz anbiguotasunak sortzen dituen arazoak eta horiek saihesteko metodoak ezagutu.
- Analizatzaile auresale errekurtsiboak eraikitzeke gai izatea, LL(1) gramatikak lortzen jakitea eta erroreetatik errekuperatzeko mekanismoak ezagutzea.
- LR analisi-prozesua ulertzea.
- Yacc erabiliaz, analizatzaile sintaktiko sinple bat eraikitzeke gai izatea

5. gaia. Exekuzio-denborako memoria kudeaketa

- 5.1 Sarrera eta esleipen estatikoa
- 5.2 Pila baten bidezko memoriaren esleipen dinamikoa
 - Aldagai ez-lokalen atzipena.
 - Parametro-pasatzea.
- 5.3 Memoriaren esleipen dinamikoa: "Heap allocation"
 - Memoriaren esleipen eta askapen esplizitu eta inplizitua.
 - Memoriaren esleipen dinamikorako teknikak.

Oinarrizko erreferentzia: 7. kapitulua [Aho, 90; Aho, 06; Louden, 04]

Erreferentzia osagarriak: 10. kapitulua [Sanchis eta Galan, 86].

Helburuak: Gai honen helburu orokorra izango da ikaslea gai izatea memoriako posizioen helbideratze-moduko makina abstraktu batentzako kodea sortzeko. Helburu hau memoriaren esleipen estatikoa zein errekursibitatea, blokeen habiaraketa edo exekuzio-denborako memoriaren esleipen dinamikoa onartzen duten lengoaiekin landuko da.

Helburu zehatzak

- Programazio-lengoaien zenbait ezaugarriren ondorioak identifikatzea, exekuzio-denborako memoriaren kudeaketaren ikuspuntutik.
- SZIE baten bidez espezifikatu ahal izatea kodearen sorkuntza memoriaren esleipen estatikoa edo pilaren bidezko esleipena onartzen duten lengoaietan.
- Lotu, blokeen habiaraketa onartzen duten lengoaiantzat, egiaztapen semantikoan egiten den sinboloen taularen kudeaketa kodearen sorkuntzaren prozesuarekin.
- Zaborraren bilketaren inplementazioa egiteko oinarrizko teknikak ezagutzea.