

DOI: <https://www.doi.org/10.36719/2706-6185/09/5-8>

**Çinarə Çingiz qızı Adilzadə**  
Azərbaycan Texnoloji Universiteti  
Gəncə, Azərbaycan  
Magistrant  
cinareadilzade@gmail.com

## PROQRAMLASHDIRMA TEXNOLOGİYALARI VƏ ALQRİTM

### Xülasə

Fortran (əvvəllər FORTRAN) ümumi təyinatlı, prosedur, imperativ proqramlaşdırma dilidir. Xüsusilə, ədədi hesablamalar və elmi hesablamalar üçün uyğundur. İlk olaraq 1950-ci illərdə IBM tərəfindən elmi və mühəndislik tətbiqləri üçün hazırlanmışdır. Fortran proqramlaşdırmanın bu sahəsinə erkən hakim oldu və rəqəmsal hava proqnozu, sonlu elementlərin təhlili, hesablama mayeləri kimi hesablama baxımından intensiv sahələrdə yarım əsrdən artıqdır ki, davamlı istifadə olunur. Bu, yüksək performanslı hesablama sahəsində ən populyar dillərdən biridir. Və dünyanın ən sürətli superkompüterlərini müqayisə edən və sıralayan proqramlar üçün istifadə olunan dildir.

**Açar sözlər:** proqramlaşdırma dili, alqoritm, Smalltalk, simulyasiya, sistem

**Chinara Chingiz gizi Adilzada**  
Azerbaijan Technological University

### Programming technologies and algorithm

### Abstract

Fortran (formerly FORTRAN) is a general-purpose, procedural, imperative programming language. In particular, it is suitable for numerical calculations and scientific calculations. It was first developed by IBM in the 1950s for scientific and engineering applications. Fortran dominated this field of programming early on and has been used continuously for more than half a century in computationally intensive fields such as digital weather forecasting, finite element analysis, and computational fluids. It is one of the most popular languages in the field of high performance computing. And it's the language used to compare and rank the world's fastest super computers.

**Keywords:** programming language, algorithm, Smalltalk, simulation, system

### Giriş

Fortran (The IBM Mathematical Formula Translating System-dən əldə edilən qarışıq) hər biri adətən əvvəlki versiyalarla uyğunluğu saxlamaqla, dilə genişlənmələr əlavə etmək üçün təkamül yolu ilə inkişaf etdirilən bir sıra versiyaları əhatə edir. Ardıcıl versiyalar xarakterə əsaslanan məlumatların emalı (FORTRAN 77), massiv proqramlaşdırma, modul proqramlaşdırma və obyekt əsaslı proqramlaşdırma (Fortran 90/95) və obyekt yönümlü və ümumi proqramlaşdırma (Fortran 2003) üçün əlavə dəstək verdi.

ALGOL (ALGOritmic Language üçün qısa) ilk olaraq 1950-ci illərin ortalarında işlənib hazırlanmış və bir çox digər dillərə böyük təsir göstərən və demək olar ki, növbəti 30 il ərzində alqoritmlərin dərsləklərdə və akademik əsərlərdə de-fakto təsvir edilməsinə çevrilmiş imperativ kompüter proqramlaşdırma dilləri ailəsidir. O, FORTRAN ilə bağlı yaranan bəzi problemlərin qarşısını almaq üçün nəzərdə tutulmuşdu və nəticədə bir çox başqa proqramlaşdırma dillərinin (BCPL, B, Paskal, Simula və C daxil olmaqla) yaranmasına səbəb oldu. ALGOL kod bloklarını təqdim etdi və onları ayırmaq üçün başlanğıc və son cütlərindən istifadə edən ilk dil oldu. Con Backus proqramlaşdırma dillərini xüsusi olaraq ALGOL 58 üçün təsvir edən Backus normal forma metodunu işləyib hazırladı. O, Peter Naur tərəfindən ALGOL 60 üçün yenidən işlənmiş və genişləndirilmiş və Donald Knutun təklifi ilə

BackusNaur Forması adlandırılmışdır. NiklausWirthPaskalın inkişafına keçməzdən əvvəl öz ALGOL W-ni ALGOL 60-a əsaslandırdı.

LISP uzun tarixə və fərqli, tam mötərizəli sintaksisə malik kompüter proqramlaşdırma dilləri ailəsidir. Əvvəlcə 1958-ci ildə təyin olunmuş Lisp bu gün geniş istifadə edilən ikinci ən qədim yüksək səviyyəli proqramlaşdırma dilidir; yalnız Fortran daha yaşlıdır. Fortran kimi, Lisp də ilk günlərindən çox dəyişib və onun tarixində bir sıra dialektlər mövcud olub. Bu gün ən çox tanınan ümumi təyinatlı Lisp dialektləri Clojure, CommonLisp və Schemedir. Lisp əvvəlcə Alonzo Kilsəsinin lambdahesablamasının qeydindən təsirlənərək kompüter proqramları üçün praktiki riyazi qeyd kimi yaradılmışdır. Tezliklə süni intellekt (AI) tədqiqatları üçün sevimli proqramlaşdırma dili oldu. Ən erkən proqramlaşdırmadillərindən biri olaraq, Lisp kompüter elmində bir çox ideyalara, o cümlədən ağac məlumat strukturları, avtomatik saxlama idarəetməsi, dinamik yazma və özünə hosting tərtibçisinə öncülük etdi. LISP adı "LİSt Emalı" sözündən götürülmüşdür. Əlaqəli siyahılar Lisp dillərinin əsas məlumat strukturlarından biridir və Lisp mənbə kodu özü siyahılardan ibarətdir. Nəticədə, Lisp proqramları proqramçılara Lisp-ə daxil edilmiş yeni sintaksis və ya hətta yeni domen-xüsusi proqramlaşdırma dilləri yaratmağa imkan verən makro sistemlərin yaranmasına səbəb olan məlumat strukturu kimi mənbə kodunu manipulyasiya edə bilər.

COBOL ən qədim proqramlaşdırmadillərindən biridir. Onun adı, şirkətlər və hökumətlər üçün biznes, maliyyə və inzibati sistemlərdə onun əsas domenini təyin edən Ümumi Biznes yönümlü Dilin abbreviaturasıdır. COBOL spesifikasiyası ilk olaraq 1959-cu ilin ikinci yarısında QreysHopper tərəfindən yaradılmışdır. Spesifikasiyalar böyük ölçüdə QreysHopper tərəfindən icad edilən və adətən "COBOL dilinin anası" adlandırılan FLOW-MATIC dilindən ilhamlanıb. 1959-cu ildən COBOL bir neçə modifikasiya və təkmilləşdirmələrə məruz qalmışdır. COBOL-un müxtəlif versiyaları arasında uyğunsuzluq problemini aradan qaldırmaq üçün Amerika Milli Standartlar İnstitutu (ANSI) 1968-ci ildə dilin standart formasını işləyib hazırladı. Bu versiya Amerika Milli Standartı (ANS) COBOL kimi tanınırdı. 1974-cü ildə ANSI 1968-ci il versiyasında olmayan bir sıra funksiyaları ehtiva edən (ANS) COBOL-un yenidən işlənmiş versiyasını nəşr etdi. 1985-ci ildə ANSI 1974-cü il standartında olmayan yeni xüsusiyyətlərə malik olan daha bir işlənmiş versiyanı nəşr etdi. Dil bu gün də inkişaf etməkdə davam edir. COBOL 2002 standartına obyekt yönümlü proqramlaşdırma və digər müasir dil xüsusiyyətləri üçün dəstək daxildir.

Simula 1960-cı illərdə Oslodakı Norveç Hesablama Mərkəzində Ole-JohanDahl və KristenNygaard tərəfindən hazırlanmış Simula I və Simula 67 adlı iki proqramlaşdırma dilinin adıdır. Sintaktik olaraq, bu, Algol 60-ın kifayət qədər etibarlı supersetidir. Simula 67 obyektləri, sinifləri, alt sinifləri, virtual metodları, koroutinləri, diskret hadisə simulyasiyasını və zibil kolleksiyasını təqdim etdi.

Simula ilk obyekt yönümlü proqramlaşdırma dili hesab olunur. Adından da göründüyü kimi, Simula simulyasiyalar etmək üçün nəzərdə tutulmuşdu və bu domenin ehtiyacları bu gün obyekt yönümlü dillərin bir çox xüsusiyyətləri üçün çərçivə təmin etdi. Simula, VLSI dizaynlarının simulyasiyası, prosesin modelləşdirilməsi, protokollar, alqoritmlər və çap, kompüter qrafikası və təhsil kimi digər proqramlar kimi geniş tətbiqlərdə istifadə edilmişdir. Simula tipli obyektlər C++, Java və C# dillərində yenidən həyata keçirildiyi üçün Simula-nın təsiri çox vaxt nəzərə çarpmır. C++ dilinin yaradıcısı BjarneStroustrup etiraf etdi ki, Simula 67 ona C++ dilini inkişaf etdirməyə, Simula tərəfindən təklif olunan məhsuldarlıq təkmilləşdirmələrini BCPL kimi aşağı səviyyəli dillərin təklif etdiyi xam hesablama sürətinə çatdırmağa ən böyük təsir göstərmişdir.

Kompüter proqramlaşdırmasında BASIC (Beginner's All-purpose Symbolic Instruction Code-un abbreviaturası) yüksək səviyyəli proqramlaşdırma dilləri ailəsidir. Orijinal BASIC 1964-cü ildə ABŞ-ın Nyu-Hempşir ştatındakı Dartmut Kollecinə JohnGeorgeKemeny və ThomasEugeneKurtz tərəfindən qeyri-elm tələbələrini kompüterə çıxışını təmin etmək üçün hazırlanmışdır. O dövrdə kompüterlərin demək olar ki, bütün istifadəsi xüsusi proqram təminatının yazılmasını tələb edirdi ki, bu da yalnız alimlərin və riyaziyyatçıların bacardığı bir şey idi. Dil və onun variantları 1970-ci illərin sonu və 1980-ci illərin sonunda mikrokompüterlərdə geniş yayıldı. Dil qismən FORTRAN II və qismən ALGOL 60-a əsaslanırdı.

Köhnə dillərin mürəkkəblik problemlərini xüsusi olaraq yeni istifadəçilər sinfi üçün yeni dil dizaynı ilə həll etmək nəzərdə tutulmuşdu ki, o zamankı yeni vaxt paylaşma sistemləri, yəni daha ənənəvi istifadəçilərin riyazi biliklərinə malik olmayan daha az texniki istifadəçiyə imkan verirdi. və onu əldə etməkdə maraqlı deyildi. Tədris və tədqiqatları dəstəkləmək üçün kompüterdən istifadə etmək o zamanlar olduqca yeni idi.

BASIC bu günə qədər Microsoft VisualBasic kimi BASIC-dən təsirlənən bir neçə yüksək dəyişdirilmiş ləhcələrdə və yeni dillərdə populyar olaraq qalır.2006-cı ilə kimi, NET platforması üçün tərtibatçıların 59%-i yeganə dil kimi VisualBasic .NET-dən istifadə edirdi.

ISWIM [9] mücərrəd kompüter proqramlaşdırma dilidir (və ya proqramlaşdırma dilləri ailəsi) Peter J. Landin tərəfindən hazırlanmış və ilk dəfə 1966-cı ildə ACM-in "Communications" jurnalında [9] nəşr olunan "TheNext 700 ProgrammingLanguages" adlı məqaləsində təsvir edilmişdir.Akronim "Nə demək istədiyimi görürsənsə" deməkdir.Tətbiq edilməməsinə baxmayaraq, proqramlaşdırma dillərinin, xüsusən SASL, Miranda, ML, Haskell və onların davamçıları kimi funksional proqramlaşdırma dillərinin inkişafında çox təsirli olduğunu sübut etdi.ISWIM-innotasiya baxımından fərqləndirici xüsusiyyəti onun harada bəndlərindən istifadə etməsidir.ISWIM proqramı harada müddəaları (dəyişənlər arasında tənliklər daxil olmaqla köməkçi təriflər), şərti ifadələr və funksiya tərifləri ilə təsnif edilən tək ifadədir.CPL ilə ISWIM, funksional və deklarativproqramlaşdırma dillərində geniş istifadə olunan wherebəndlərindən istifadə edən ilk proqramlaşdıradillərindən biri idi.

Paskal 1968/9-cu illərdə tərtib edilmiş və 1970-ci ildə NiklausWirth tərəfindən strukturlaşdırılmışproqramlaşdırma və verilənlərinstrukturlaşdırılmasından istifadə edərək yaxşı proqramlaşdırma təcrübələrini təşviq etmək üçün kiçik və səmərəli dil kimi nəşr edilmiş təsirli imperativ və prosedurproqramlaşdırma dilidir.Paskal dili ALGOL proqramlaşdırma dilinə əsaslanır və fransız riyaziyyatçısı və filosofu BlezPaskalın şərəfinə adlandırılıb.Wirth daha sonra Paskal dilinə oxşar olan Modula-2 və Oberon dillərini inkişaf etdirdi.Paskaldan əvvəl və ondan əvvəl WirthEuler dilini inkişaf etdirdi, ardınca Algol-W.Başlanğıcda Paskal əsasən tələbələrə strukturlaşdırılmışproqramlaşdırmanı öyrətmək üçün nəzərdə tutulmuşdu.Tələbələrin nəsilləri Paskaldan bakalavr kurslarında giriş dili kimi istifadə ediblər.ObjectPascal kimi tanınan törəmə obyekt yönümlüproqramlaşdırma üçün nəzərdə tutulmuşdur.

Smalltalk obyekt yönümlü, dinamik tipli, əks etdirən proqramlaşdırma dilidir.Smalltalk, "insan kompüter simbiozu" ilə misal olaraq hesablamının "yeni dünyasını" dəstəkləmək üçün dil kimi yaradılmışdır. O, 1970-ci illərdə Alan Kay, Dan İnqalls, AdeleQoldberq, TedKaehler, SkottUolles və başqaları tərəfindən Xerox PARC-nin Öyrənmə Tədqiqat Qrupunda təhsil məqsədləri üçün nəzərdə tutulmuş və qismən yaradılmışdır.

Dil ilk olaraq Smalltalk-80 olaraq buraxıldı və o vaxtdan bəri geniş istifadə olunur.Smalltalk kimi dillər fəal inkişafdadır və öz ətrafında sadıq istifadəçilər icmalarını toplayıb. ANSI Smalltalk 1998-ci ildə ratifikasiya olunub və ilkin versiyalardan biri olan Smalltalk-76-nın standart versiyasını təmsil edir, sinif kitabxanası kodu da daxil olmaqla, indi tanış olan alətlərin əksəriyyətini özündə əks etdirən inkişaf mühiti ilə birlikdə həyata keçirilən ilk proqramlaşdıradillərindən biri idi. Smalltalk-80 xassələri və davranışı fərdi siniflərlə əlaqələndirərək "hər şey obyektidir" (şəxsi nümunə dəyişənləri istisna olmaqla) paradigmasını qorumağa kömək etmək üçün metasiniflər əlavə etdi.

Smalltalk dörd əsas sahədə kompüter proqramlaşdırmasının daha geniş dünyasına təsir etdi. Bu, digər kompüter proqramlaşdırma dillərinin sintaksisi və semantikasına ilham verdi. İkincisi, o, mesaj ötürülməsi kimi tanınan hesablama modelinin prototipi idi. Üçüncüsü, onun WIMP GUI-i iyirminci əsrin sonu və iyirmi birinci əsrin əvvəllərində fərdi kompüterlərin pəncərə mühitini ilhamlandırırdı, o qədər ki, ilk Macintosh iş masasının pəncərələri Smalltalk-80-in MVC pəncərələri ilə demək olar ki, eyni görünür. Nəhayət, inteqrasiya olunmuş inkişaf mühiti Smalltalk-ın kod brauzerləri və sazlayıcılarına bənzəyən vizual proqramlaşdırma vasitələri nəslü üçün model idi.

### Nəticə

Beləliklə, digər obyekt yönümlü dillərdə olduğu kimi, Smalltalk-80-də də (lakin Smalltalk-72-də deyil) mərkəzi anlayış obyektidir. Obyekt həmişə bir sinif nümunəsidir. Siniflər nümunələrinin xassələrini və davranışlarını təsvir edən “planlar”dır. Məsələn, Window sinfi pəncərələrin etiket, mövqe və pəncərənin görünüb-görünməməsi kimi xassələrə malik olduğunu bildirəcək. Sinif həmçinin nümunələrin açma, bağlama, köçürmə və gizlənmə kimi əməliyyatları dəstəklədiyini bəyan edərdir. Hər bir xüsusi Pəncərə obyektini bu xassələrin öz dəyərlərinə malik olacaq və onların hər biri öz sinfi ilə müəyyən edilmiş əməliyyatları yerinə yetirə biləcək.

Smalltalk obyektini üç şeyi edə bilər:

1. Saxlama vəziyyəti (digər obyektlərə istinadlar).
2. Özündən və ya başqa obyektə mesaj alın.
3. Mesajın işlənməsi zamanı özünə və ya başqa obyektə mesaj göndərin.

Bir obyektin saxladığı vəziyyət həmişə həmin obyekt üçün özəldir. Digər obyektlər yalnız bunu etmək üçün obyektə sorğular (mesajlar) göndərməklə bu vəziyyəti sorğulaya və ya dəyişə bilər. İstənilən mesaj istənilən obyektə göndərilə bilər: mesaj alındıqda qəbulədiyi həmin mesajın uyğun olub olmadığını müəyyən edir.

### Ədəbiyyat

1. Jason Allen, Scott Collison, Robin Luckey və Andy Verprauskus. 2008.
2. Francois Bourdoncle və Stephan Merz. *Funksional proqramlaşdırma, sinif əsaslı obyekt yönümlü proqramlaşdırma və multi-metodların inteqrasiyası haqqında*. Texniki hesabat, Centre de Mathematiques Appliquees, Ecole des Mines de Paris və Institut für Informatik, Technische Universität München, oktyabr 1996-cı il
3. Robert G. Clark. *Müqayisəli proqramlaşdırma dilləri*. Addison-Wesley, 3-cü nəşr, noyabr 2000.
4. Dale Green. Java əks etdirmə API. Sun Microsystems, Inc., 2001–2005.
5. P.J. Landin. *Növbəti 700 proqramlaşdırma dili*. *ACM-nin kommunikasiyaları*, 9(3):157-166, 1966.
6. Kenneth C. Loudon. *Kompilyatorun qurulması: Prinsiplər və Təcrübə*. PWS Nəşriyyat Şirkəti, 1997.
7. Lee E. McMahon, Paolo Bonzini, Aur'elio M. Jargas, Eric Pement, Tilmann Bitterberg, Yao-Jen Chang, Yiorgos Adamopoulos, və başqaları – mətnin filtrlənməsi və dəyişdirilməsi üçün axın redaktoru. May 2008.
8. Brian McNamara və Yannis Smaragdakis. FC++ kitabxanasından istifadə edərək C++ dilində funksional proqramlaşdırma. 2001.
9. Tobias Nipkow, Lawrence C. Paulson və Markus Wenzel. Isabelle/HOL: *Daha yüksək səviyyəli məntiq üçün sübut köməkçisi*, cild 2283. Springer-Verlag, dekabr 2007.
10. Joey Paquet. COMP6411 üçün kurs qeydləri, qış 2010. *Kompüter Elmləri və Proqram Mühəndisliyi Departamenti*, Konkordiya Universiteti, Monreal, Kanada, 2010.

**Rəyçi: dos. Süleymanov Kamil Mursəl oğlu**

Göndərilib: 05.03.2022

Qəbul edilib: 25.03.2022