

Հետազոտական  
աշխատանք:  
Ծրագրավորման  
լեզուների  
ստեղծումն ու  
զարգացումը

- **Նպատակը:** Ծանոթացնել մարդկանց ծրագրավորման լեզուների հետ:

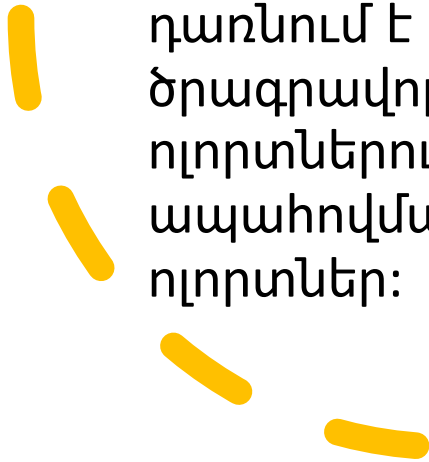
### Բովանդակություն

- Ներածություն
- Աստեմբլեր ծրագրավորման լեզվի մասին
- Երկրորդ կարգի ծրագրավորման լեզուներ
- Բարձր կարգի ծրագրավորման լեզուներ
- Ծրագրավորման լեզուների անվանումներ
- Շուկայում պահանջվող լեզուներ
- Ծրագրավորողի մասնագիտության մասին
- Եզրակացություն



## Ներածություն

- Հետազոտական աշխատանքում ներկայացնում եմ ծրագրավորման լեզուները: Ընտրել եմ այս թեման, քանի որ ծրագրավորում ինձ հետաքրքիր է և ես ուզում եմ, որ նրա մասին բոլորը ավելի շատ իմանան: Ծրագրավորման լեզուներին տիրապետելը օգնում է պահանջված մասնագիտություն ձեռք բերել, թույլ է տալիս խնդիրների լուծմանը մոտենալ տարբեր տեսանկյուններից, ինչը շատ կարևոր է 21-րդ դարում: Ժամանակակից աշխարհում ծրագրավորման լեզուների իմացությունը դառնում է ավելի ու ավելի տարածված մասնագիտություն: Առանց ծրագրավորման հմտությունների դժվարանում են կառավարել բազմաթիվ ոլորտներում, ինչպիսիք են տեղեկատվական տեխնոլոգիաները, ծրագրային ապահովման մշակումը, գիտական հետազոտությունները, ֆինանսները և շատ այլ ոլորտներ:



## Ասսեմբլեր ծրագրավորման լեզվի մասին

Իմ պրոեկտը սկսելուց առաջ ես կցանկանամ պատմել, թե ինչ է ասսեմբլերը, քանի որ ոչ բոլորը գիտան դրա մասին: Ասսեմբլերը ցածր մակարդակի ծրագրավորման լեզու է: Ասսեմբլերը թույլ է տալիս գրել կարճ և արագ կատարվող ծրագրեր, մեծ մասամբ ասսեմբլերով գրվում են այնպիսի ծրագրեր, որոնք ապահովում են մեքենայական մասի արդյունավետությունը:

Ասսեմբլեր ծրագրավորման լեզվով աշխատում է համակարգչի BIOS համակարգը: Խոսել բիոսի մասին, ինչ է, ինչի համար է պետք, առաջ համակարգիչները միայն բիոսով էին աշխատում, նման աշխատանքի առավելություններն ու թերությունները:



## Առավելությունները

Ասսեմբլեր լեզուն թույլ է տալիս գրել ամենաարագ և փոքր ծրագիր, որը ընդհանրապես հնարավոր է գրել տվյալ պրոցեսորի համար:

Ասսեմբլեր լեզվի օգնությամբ ստեղծվում են մեքենայից կախված ենթածրագրեր նախաթարգմանիչի համար:

Ասսեմբլեր լեզուն օգտագործվում է BIOS-ի համար հանգույցներ ստեղծելիս:

Ասսեմբլեր լեզուն շատ հաճախ օգտագործվում է սարքերի դրայվերների ստեղծման ժամանակ:

## Թերությունները

Աստեմբլերը ցածր մակարդակի ծրագրավորման լեզու է, ուստի նրանով գրված ծրագրերը ավելի բարդ են ընթերցվում և դժվար է հասկանալ, քան բարձր մակարդակի ծրագրավորման լեզվով գրված ծրագրերը:

Ծրագիրը կազմված է շատ «փոքր» տարրերից՝ մեքենայական հրամաններից, որոնք բարդեցնում են ծրագրավորման գործընթացը, հաճախացնում սխալվելու հավանականությունը:

Բացակայում է ծրագրի տեղափոխության հնարավորությունը մի համակարգից դեպի այլ համակարգիչ:

Ինչքանո՞վ է արդարացված մինչև հիմա ասսեմբլերով ծրագրավորումը, ինչու՞ են դեռ օգտագործում, արդյո՞ք հնարավոր չէ փոխարինել այլ լեզվով և ինչու՞:

- Եթե, որպես ծրագրավորման լեզվի առաջին սերունդ համարենք մեքենայական կոդը, ապա ասսեմբլեր լեզուն կարելի է համարել ծրագրավորման լեզուների երկրորդ սերունդ: Ասսեմբլեր լեզվի մի շարք թերությունները բերեցին ծրագրավորման լեզուների երրորդ սերնդի առաջացմանը, որոնք կոչվեցին բարձր մակարդակի ծրագրավորման լեզուներ: Հենց բարձր մակարդակի ծրագրավորման լեզուները և իրենց ժառանգներն են, որ հիմնականում օգտագործվում են ներկայիս ինֆորմացիոն տեխնոլոգիաների արդյունաբերությունում:
- Ասսեմբլեր լեզվով գրվում են ծրագրեր կամ ծրագրի հատվածներ այն դեպքում, երբ հատկապես խիստ անհրաժեշտ է արագագործությունը: Սա է հիմնական պատճառը, որ լեզուն դեռ կիրառվում է:

# Հրամանների համակարգ Ասամբլեր ծրագրավորման լեզվում

- Ասամբլեր լեզվում հրամանների համակարգը բաժանվում է հետևյալ հիմնական խմբերի`
- Տվյալների տեղափոխման հրամաններ (`mov`, `push` և այլն)
- Թվաբանական հրամաններ (`add`, `inc` և այլն)
- Տրամաբանական հրամաններ (`or`, `and` և այլն)
- Ղեկավարությունը փոխանցող հրամաններ (`jmp`, `loop` և այլն)
- Ընդհատման հրամաններ (`int` և այլն)

# Կոդի օրինակ

```
C000          ORG    ROM+$0000 BEGIN MONITOR
C000 8E 00 70  START  LDS    #STACK

*****
* FUNCTION: INITA - Initialize ACIA
* INPUT: none
* OUTPUT: none
* CALLS: none
* DESTROYS: acc A

0013          RESETA EQU    %00010011
0011          CTLREG EQU    %00010001

C003 86 13          INITA  LDA  A  #RESETA  RESET ACIA
C005 B7 80 04          STA  A  ACIA
C008 86 11          LDA  A  #CTLREG  SET 8 BITS AND 2 STOP
C00A B7 80 04          STA  A  ACIA

C00D 7E C0 F1          JMP    SIGNON  GO TO START OF MONITOR

*****
* FUNCTION: INCH - Input character
* INPUT: none
* OUTPUT: char in acc A
* DESTROYS: acc A
* CALLS: none
* DESCRIPTION: Gets 1 character from terminal

C010 B6 80 04  INCH   LDA  A  ACIA      GET STATUS
C013 47          ASR  A          SHIFT RDRF FLAG INTO CARRY
C014 24 FA          BCC  INCH     RECIEVE NOT READY
C016 B6 80 05          LDA  A  ACIA+1  GET CHAR
C019 84 7F          AND  A  #$7F    MASK PARITY
C01B 7E C0 79          JMP    OUTCH   ECHO & RTS

*****
* FUNCTION: INHEX - INPUT HEX DIGIT
* INPUT: none
* OUTPUT: Digit in acc A
* CALLS: INCH
* DESTROYS: acc A
* Returns to monitor if not HEX input

C01E 8D F0          INHEX  BSR    INCH    GET A CHAR
C020 81 30          CMP  A  #'0    ZERO
C022 2B 11          BMI  HEXERR  NOT HEX
C024 81 39          CMP  A  #'9    NINE
C026 2F 0A          BLE  HEXRTS  GOOD HEX
C028 81 41          CMP  A  #'A
C02A 2B 09          BMI  HEXERR  NOT HEX
C02C 81 46          CMP  A  #'F
C02E 2E 05          BGT  HEXERR
C030 80 07          SUB  A  #7
C032 84 0F          HEXRTS AND  A  #$0F  CONVERT ASCII TO DIGIT
C034 39          RTS

C035 7E C0 AF          HEXERR JMP    CTRL    RETURN TO CONTROL LOOP
```



# Երկրորդ կարգի ծրագրավորման լեզուներ

- Ասամբլեային լեզուները ծրագրավորման լեզուների միակ տեսակն են, որը պատկանում է երկրորդ մակարդակի ծրագրավորման լեզուների կատեգորիային:
- **1.FORTRAN Ասամբլեային լեզու (FAL).** Ասամբլեայի լեզու է, որը հատուկ նախագծված է FORTRAN ծրագրերի համար, որն ապահովում է ավելի կառուցվածքային մոտեցում ասամբլի լեզվով ծրագրավորման համար: Այն թույլ է տալիս հիշողության և ռեգիստրների անմիջական մանիպուլյացիա՝ դարձնելով այն պիտանի FORTRAN-ի կատարողականի համար, կարևոր հավելվածների համար:
- **2.IBM Assembler Language (BAL).** IBM-ի հիմնական համակարգիչների հավաքման լեզուն է, որը լայնորեն օգտագործվում է համակարգի ծրագրավորման և կատարողականի համար կարևոր հավելվածներում: Այն առաջարկում է հզոր հնարավորություններ՝ կառավարելու հիշողությունը, ռեգիստրները և մուտք/ելք գործողությունները:
- **3.MASM (Microsoft Macro Assembler).** հավաքման լեզու է, որը մշակվել է Microsoft-ի կողմից x86 և x64 ճարտարապետությունների համար: Այն հայտնի է մակրոների, պայմանական հավաքման և առաջադեմ հասցեավորման ռեժիմների աջակցությամբ:
- **4.TASM (Turbo Assembler).** հավաքման լեզու է, որը մշակվել է Borland-ի կողմից x86 և x64 ճարտարապետությունների համար: Այն հանրաճանաչ ընտրություն է սարքի դրայվերների, ցածր մակարդակի համակարգի ծրագրային ապահովման և կատարողականի օպտիմալացված կոդ գրելու համար:
- **5.NASM (Netwide Assembler).** դյուրակիր հավաքման լեզու է, որը կարող է օգտագործվել տարբեր ճարտարապետությունների համար, ներառյալ x86, x64, ARM և MIPS: Այն հայտնի է իր միջպլատֆորմային համատեղելիությամբ և տարբեր գործառույթների աջակցությամբ, ինչպիսիք են մակրոները և պայմանական հավաքումը:
- **6.GAS (GNU Assembler).** GNU Compiler Collection-ի (GCC) մի մասն է, GAS-ը շարժական հավաքման լեզու է, որն աջակցում է տարբեր ճարտարապետությունների, ներառյալ x86, x64, ARM և MIPS: Այն հայտնի է իր ճկունությամբ և GNU-ի այլ գործիքների հետ համատեղելիությամբ:
- **7.HLA (Hewlett-Packard Assembler).** PA-RISC ճարտարապետությունների համար հավաքման լեզու է մշակված Hewlett-Packard-ի կողմից: Այն ապահովում է հզոր առանձնահատկություններ՝ հիշողության, ռեգիստրների և մուտք/ելք գործողությունների կառավարման համար:
- **8.Motorola Assembly Language (M68k).** Մոտորոլայի 68000 սերիայի պրոցեսորների հավաքման լեզու: Այն լայնորեն օգտագործվում էր ներկառուցված համակարգերում և վաղ անհատական համակարգիչներում:
- **9.Z80 Assembly Language.** Z80 միկրոպրոցեսորի հավաքման լեզու, որն օգտագործվում էր տարբեր տնային համակարգիչներում, խաղային վահանակներում և ներկառուցված համակարգերում:

# Բարձր կարգի ծրագրավորման լեզուներ

- 1.Python.** Ընդհանուր նշանակության ծրագրավորման լեզու է, որը հայտնի է իր պարզությամբ, ընթեռնելիությամբ և գրադարանների ու գործիքների լայն տեսականիով: Այն լայնորեն օգտագործվում է վեբ ծրագրավորման, տվյալագիտության, մեքենայական ուսուցման և գիտական հաշվարկների համար:
- 2.Java.** Բազմակողմանի և օբյեկտին ուղղված ծրագրավորման լեզու է, որը հայտնի է իր միջավայրային համատեղելիությամբ և կայունությամբ: Այն լայնորեն օգտագործվում է ձեռնարկատիրական ծրագրերի, վեբ ծրագրավորման, բջջային հավելվածների մշակման և ներկառուցված համակարգերի համար:
- 3.JavaScript.** Սկրիպտավորման լեզու է, որը հիմնականում օգտագործվում է վեբ մշակման և front-end ծրագրավորման համար: Այն նաև օգտագործվում է սերվերի կողմից ծրագրավորման համար JavaScript շրջանակներով, ինչպիսին է Node.js-ը:
- 4.C/C++.** Հզոր և արդյունավետ ծրագրավորման լեզուներ են, որոնք ապահովում են ուղղակի ապարատային կառավարում և ցածր մակարդակի հիշողության կառավարում: Դրանք լայնորեն օգտագործվում են համակարգի ծրագրավորման, օպերացիոն համակարգերի մշակման, խաղերի մշակման և կատարողականի համար կարևոր հավելվածների համար:
- 5.C#.** Microsoft-ի կողմից մշակված ընդհանուր նշանակության ծրագրավորման լեզու է: Այն հայտնի է իր օբյեկտի վրա հիմնված մոտեցմամբ, ուժեղ մուտքագրումներով և Microsoft-ի .NET շրջանակի հետ ինտեգրմամբ:
- 6.PHP.** Սկրիպտավորման լեզու է, որը հիմնականում օգտագործվում է վեբ կայքերի մշակման համար, հատկապես սերվերի կողմից գրելու համար: Այն հայտնի է իր հեշտ օգտագործման և վեբ ծրագրավորման շրջանակներում լայն կիրառումով:
- 7.R.** Ծրագրավորման լեզու է, որը հատուկ նախագծված է վիճակագրական հաշվարկների և տվյալների վերլուծության համար: Այն լայնորեն օգտագործվում է ակադեմիայում, արդյունաբերության մեջ և հետազոտություններում տվյալների գիտության, մեքենայական ուսուցման և վիճակագրական մոդելավորման համար:
- 8.Kotlin.** JetBrains-ի կողմից մշակված ժամանակակից և հակիրճ ծրագրավորման լեզու է: Այն հայտնի է Java-ի հետ իր փոխգործունակությամբ, անվտանգության առանձնահատկություններով և Android հավելվածներ մշակելու համար հարմարությամբ:
- 9.Swift.** Ծրագրավորման լեզու է, որն օգտագործվում է iOS, macOS, watchOS և tvOS հավելվածներ մշակելու համար: Այն հայտնի է իր ժամանակակից շարահյուսությամբ, անվտանգության առանձնահատկություններով և զարգացման հզոր գործիքներով:
- 10.Go.** Google-ի կողմից մշակված միաժամանակյա ծրագրավորման լեզու է: Այն հայտնի է իր պարզությամբ, կատարողականությամբ և օգտագործման հեշտությամբ՝ բաշխված համակարգերի և ցանցային հավելվածների կառուցման համար:



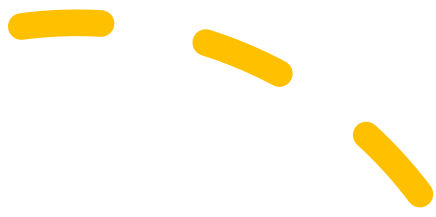
# Ծրագրավորման լեզուների անվանումներ

- Ընդհանուր նշանակության ծրագրավորման լեզուներ.
- Python
- Java
- C
- C++
- JavaScript
- C#
- PHP
- Go
- Kotlin
- Swift
- Ruby
- Rust
- Haskell



```
Try Kotlin
Simplest version  An Object-oriented Hello

class Greeter(val name) {
    fun greet() {
        println("Hello")
    }
}
```



## Վեր մշակման ծրագրավորման լեզուներ.

- Front-end լեզուները: HTML, CSS, JavaScript
- Back-end լեզուները: Python, Java, PHP, Ruby, JavaScript (Node.js)
- Full-stack լեզուները: JavaScript (React, Angular, Vue), TypeScript

## Տվյալների գիտություն և մեքենայական ուսուցման ծրագրավորման լեզուներ.

- Python
- R
- Julia
- MATLAB
- SAS

```
<?php global $speak; >
<!DOCTYPE html >
<html <?php language_attributes(); ?>
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />
    <?php wp_head(); ?>
</head>
<body <?php body_class(); ?>
<div class="navbar navbar-default menu-back">
    <div class="container">
        <div class="navbar-header">
            <button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <a class="navbar-brand" href=" <?php echo home_url(); ?>">
                
            </a>
        </div>
        <div class="navbar-collapse collapse">
            <?php
            wp_nav_menu(array(
                'menu_class' => 'nav navbar-nav navbar-right',
                'theme_location' => 'primary',
                'container' => false,
                'walker' => new custom_nav_menu(),
            ));
        </div>
    </div>
</div>
<!-- Top Menu End -->
<?php if($speak['show_hide_social'] == '0') : ?>
<div class="div-social-top">
    <?php if(isset($speak['header_social_email']) && !empty($speak['header_social_email'])) : ?>
        <i class="fa fa-globe"></i>E-mail: <?php echo $speak['header_social_email']; ?>
    </div>
</div>
</div>
```

## Բջջային հավելվածների մշակման ծրագրավորման լեզուներ.

- Android-ի բնիկ զարգացում: Java, Kotlin
- iOS-ի բնիկ զարգացում: Swift
- Բջջային հավելվածների բազմալատֆորմային մշակում: Flutter, React Native, Xamarin

## Խաղերի մշակման ծրագրավորման լեզուներ.

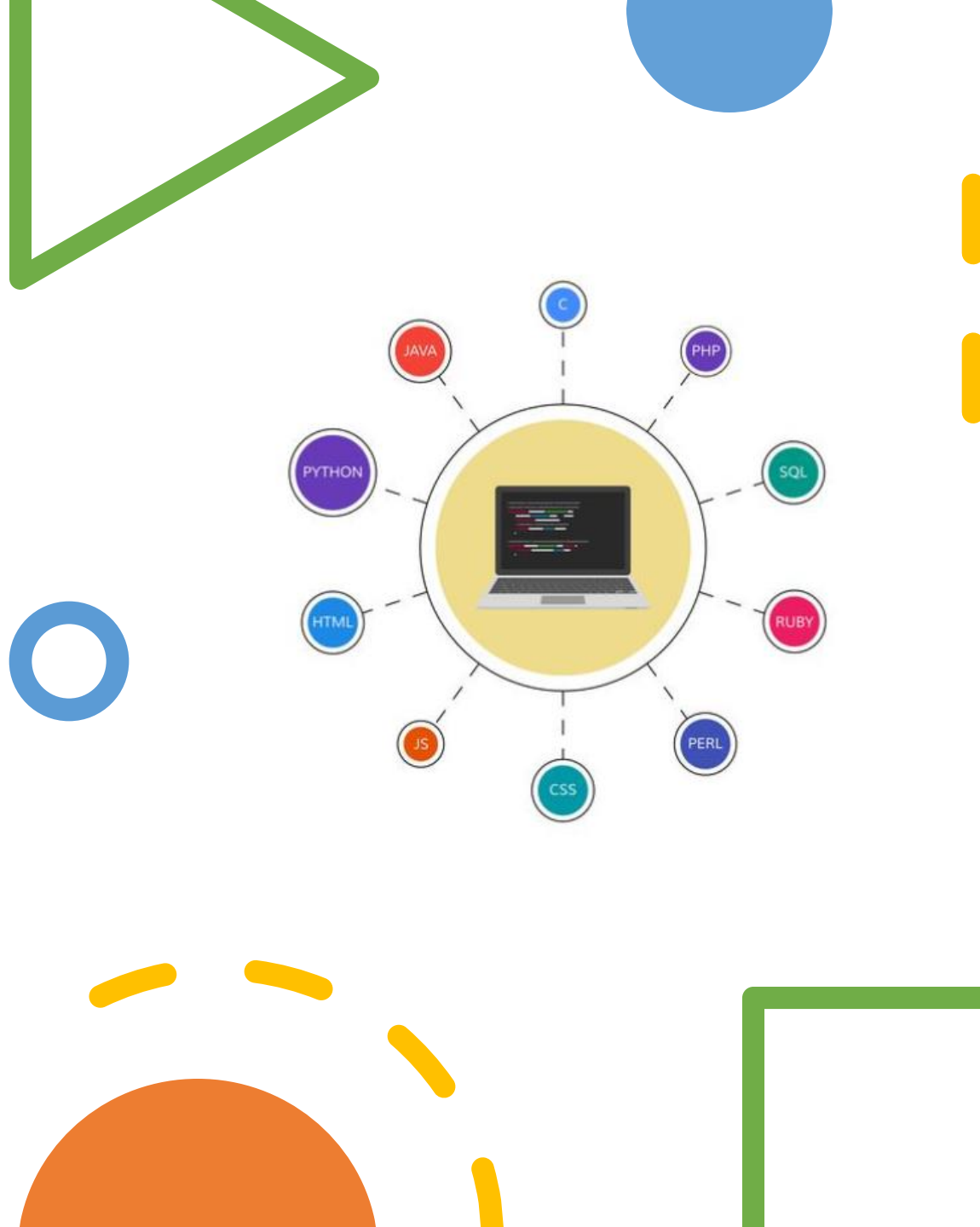
- C++
- C# (Unity)
- Java (Android խաղեր)
- JavaScript (HTML5 խաղեր)
- Scratch

## Ներկառուցված համակարգերի ծրագրավորման լեզուներ.

- C
- C++
- Assembly լեզուն

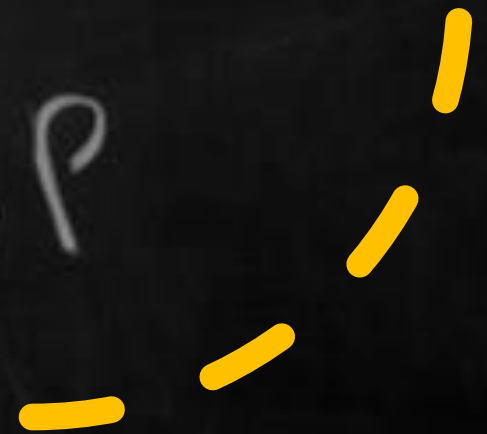
## Ծրագրավորման այլ լեզուներ՝ հատուկ նիշերով.

- SQL (տվյալների բազայի կառավարում)
- Prolog (տրամաբանական ծրագրավորում)
- Lisp (ֆունկցիոնալ ծրագրավորում)
- Erlang (համաժամանակյա ուղղվածությամբ ծրագրավորում)



# Շուկայում պահանջվող լեզուներ

- Python
- JavaScript
- Java
- C/C++
- TypeScript
- Go
- Kotlin
- Swift
- PHP
- R



# Ծրագրավորողի մասնագիտության մասին

## Ո՞վ է ծրագրավորողը

- Անձ, ով ստեղծում է համակարգչային ծրագրեր, կայքեր, վեբ ծառայություններ և բջջային հավելվածներ՝ օգտագործելով ծրագրավորման լեզուները՝ Python, C, C++, Go, Java, JS, Swift և այլն: Ծրագրավորողը թիմում աշխատում է այլ ծրագրավորողների և դիզայներների հետ՝ նախագծելու հավելվածի կամ ծրագրաշարի յուրաքանչյուր մաս, այնուհետև որոշելու, թե ինչպես են նրանք աշխատելու միասին:

## Ի՞նչ է անում

- Ծրագրավորման լեզուներով կոդ է գրում կայքերի, ծրագրերի, արհեստական ինտելեկտի, օպերացիոն համակարգերի համար:

## Ինչպիսի՞ ապագա ունի

- Ծրագրավորողները դեռ երկար ժամանակ կպահանջվեն, քանի որ կան ծրագրավորման տարբեր ճյուղեր, որոնցից մեկն էլ արհեստական ինտելեկտն է, որը պետք է անընդհատ վերապատրաստվի և կատարելագործվի, ինչպես նաև պետք է ուղղվեն դրա սխալները և թերությունները շտկվեն:



# Եզրակացություն

---

- Եզրակացնելով ասեմ, որ ծրագրավորումը ապագայում շատ կարևոր դեր կունենա մեր կյանքում: Կարծում եմ՝ ապագայում ծրագրավորումը կլինի սովորական դպրոցական առարկաների մակարդակի մի բան և բոլորը դա կիմանան: Իսկ հիմա մենք պետք է ուսումնասիրենք այն, որպեսզի հետագայում մեզ համար ավելի հեշտ լինի հարմարվել արագ փոփոխություններին: Այսօր հատկապես կարևոր է սկսել ծրագրավորում սովորելը, քանի որ դա մեզ թույլ կտա ավելի հեշտությամբ հարմարվել արագ փոփոխվող տեխնոլոգիաներին: Տեղեկատվական տեխնոլոգիաների աշխարհը զարգանում է անհավատալի արագությամբ, և մեր ժամանակակից հասարակության մեջ հաջողակ լինելու համար անհրաժեշտ է ունենալ ծրագրավորման տարրական հմտություններ: Նրանք կօգնեն մեզ ավելի լավ հասկանալ մեզ շրջապատող աշխարհը, բարելավել մեր աշխատանքի որակը և իրականացնել մեր սեփական գաղափարներն ու նախագծերը: