

«ԾՐԱԳՐԱՎՈՐՄԱՆ ՀԻՄՈՒՆՔՆԵՐ» դասընթաց

այլ ոլորտներից դեպի տեխնոլոգիական ոլորտ
սկսնակների համար



edu2020.am

ԴԱՍ #5



ՀԱՅ-ՌՈՒՄԱԿԱՆ
ՀԱՄԱԼՍԱՐԱՆ



ՀԱՅԱՍՏԱՆԻ ՀԱՆՐԱՊԵՏՈՒԹՅԱՆ
ԲԱՐՁՐ ՏԵԽՆՈԼՈԳԻԱԿԱՆ
ԱՐԴՅՈՒՆԱԲԵՐՈՒԹՅԱՆ ՆԱԽԱՐԱՐՈՒԹՅՈՒՆ

while ցիկլ

Քանի դեռ **expression**-ի արժեքը *false կամ 0* չէ կատարել **statement1**, **statement2**,
...

```
while(expression) {  
    statement1;  
    statement2;  
    ....  
}
```

```
#include <iostream>  
int main() {  
    int i = 0;  
    while(i < 10) {  
        std::cout << i << std::endl;  
        i = i + 1;  
    }  
}
```

<https://repl.it/@SevakRAU/RAUWhileEx1>



0
1
2
3
4
5
6
7
8
9



while ցիկլ

Քանի դեռ **expression**-ի արժեքը *false կամ 0* չէ կատարել **statement1**, **statement2**,
...

```
while(expression) {  
    statement1;  
    statement2;  
    ....  
}
```

```
#include <iostream>  
int main() {  
    int i = 0;  
    while(i < 10) {  
        std::cout << i << std::endl;  
        i = i + 1;  
    }  
}
```

<https://repl.it/@SevakRAU/RAUWhileEx1>



0
1
2
3
4
5
6
7
8
9



while ցիկլ, օրինակ

```
#include <iostream>
int main() {
    int i = 0;
    while(i < 10) {
        std::cout << i << std::endl;
        i = i + 1;
    }
}
```

Ամբողջ տիպի փոփոխական,
որի սկզբնական արժեքը 0 է

<https://repl.it/@SevakRAU/RAUWhileEx1>



while ցիկլ, օրինակ

```
#include <iostream>
int main() {
    int i = 0;
    while(i < 10) {
        std::cout << i << std::endl;
        i = i + 1;
    }
}
```

Քանի դեռ i փոփոխականի
արժեքը փոքր է 10-ից,
կատարել ցիկլի մարմինը

<https://repl.it/@SevakRAU/RAUWhileEx1>



while ցիկլ, օրինակ

```
#include <iostream>
int main() {
    int i = 0;
    while(i < 10) {
        std::cout << i << std::endl;
        i = i + 1;
    }
}
```

<https://repl.it/@SevakRAU/RAUWhileEx1>

Տպել i փոփոխականի
արժեքը

Ավելացնել i փոփոխականի
արժեքը 1-ով



Անվերջ ցիկլի օրինակ

```
#include <iostream>
int main() {
    int i = 0;
    while(i < 10) {
        std::cout << i << std::endl;
        // i = i + 1;
    }
}
```

<https://repl.it/@SevakRAU/RAUWhileEx1>

*i փոփոխականի արժեքը
միշտ փոքր է 10-ից*



Օրինակ. անհավասարության պայման

```
// RAU code example
#include <iostream>
int main() {
    int i = 0;
    while(i != 10) {
        std::cout << i << std::endl;
        i = i + 1;
    }
}
```



0
1
2
3
4
5
6
7
8
9

<https://repl.it/@SevakRAU/WhileEqualCond>



Օրինակ. ինդեքսի նվազեցում

// RAU code example

```
#include <iostream>
```

```
int main() {
```

```
    int i = 10;
```

```
    while(i >= 0) {
```

```
        std::cout << i << std::endl;
```

```
        i = i - 2;
```

```
    }
```

```
}
```



10
8
6
4
2
0

<https://repl.it/@SevakRAU/WhileDecIndex>



Օրինակ. պայմանների միավորում

```
#include <iostream>
```

```
int main() {  
    int x, y, i = 0, j = 0;  
    std::cin >> x;  
    std::cin >> y;  
    while (i < x && j <= y) {  
        std::cout << "i and j values are " << i << " " << j << std::endl;  
        i += 1;  
        j += 2;  
    }  
}
```

Կարելի է մի քանի պայմաններ
միավորել տրամաբանական
գործողություններով

<https://repl.it/@SevakRAU/WhileComplexCond>



Խնդիրներ

1. Մուտքագրել X ամբողջ թիվ $[0, 100]$ միջակայքից: Տպել $[0, X]$ միջակայքի բոլոր ամբողջ թվերը աճման/նվազման կարգով:
2. Մուտքագրել X ամբողջ թիվ $[0, 20]$ միջակայքից: Տպել $[1, X]$ միջակայքի բոլոր ամբողջ թվերի գումարը/արտադրյալը:
3. Մուտքագրել N հատ թիվ և տպել **YES**, եթե բոլոր թվերի գումարը գույգ է և **NO** հակառակ դեպքում



if/else օպերատորները ցիկլերում

Ցիկլերում կարելի է օգտագործել *if/else* ինչպես ծրագրի մնացած մասերում:

```
#include <iostream>
```

```
int main() {  
    int x, i = 0;  
    std::cin >> x;  
    while (i <= x) {  
        if (i % 2 == 0) {  
            std::cout << "even number - " << i << std::endl;  
        }  
        i += 1;  
    }  
}
```

Տպում է $[0, x]$ -ի
բոլոր զույգ թվերը

<https://repl.it/@SevakRAU/WhileIf>



Օրինակ. գտնել մեծագույն թիվ

Գտնել մուտքագրված 10 ամբողջ թվերից մեծագույնը/**փոքրագույնը**:

```
#include <iostream>
#include <limits.h>

int main() {
    int i = 0, x, max = INT_MIN;
    while (i < 10) {
        std::cin >> x;
        if (x > max) {
            max = x;
        }
        i += 1;
    }
    std::cout << "Max is " << max << std::endl;
}
```

<https://repl.it/@SevakRAU/MaxNumber>



Խնդիրներ

1. Մուտքագրել 10 թիվ, ամեն թիվը մուտքագրելուց տպել ***EVEN*** եթե թիվը գույգ է ***ODD*** հակառակ դեպքում:
2. Մուտքագրել X ամբողջ թիվ $[0, 20]$ միջակայքից: Տպել $[1, X]$ միջակայքի բոլոր գույգ թվերի գումարը և կենտ թվերի արտադրյալը:
3. Հաշվել մուտքագրված ամբողջ թվի թվանշանների քանակը:
4. Հաշվել մուտքագրված ամբողջ թվի թվանշանների գումարը:



break հրաման

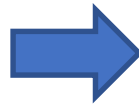
- **break** հրամանը թույլ է տալիս ընդհատել ցիկլի աշխատանքը
- **break** հրամանից հետո ղեկավարումը փոխանցվում է ցիկլի անմիջապես հաջորդ հրամանին



Օրինակ

```
#include <iostream>
int main() {
    int i = 0;
    while (i <= 10) {
        std::cout << i << std::endl;
        i = i + 1;
        if (i > 5) {
            break;
        }
    }
    std::cout << "Loop end";
}
```

Եթե **i** փոփոխականի
արժեքը մեծ է **5-ից**
դադարեցնել ցիկլի
կատարումը



0
1
2
3
4
5
Loop end

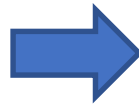
<https://repl.it/@SevakRAU/WhileBreak>



Օրինակ

```
#include <iostream>
int main() {
    int i = 0;
    while (i <= 10) {
        std::cout << i << std::endl;
        i = i + 1;
        if (i > 5) {
            break;
        }
    }
    std::cout << "Loop end";
}
```

Ընդհատել ցիկլի
կատարումը



0
1
2
3
4
5
Loop end

<https://repl.it/@SevakRAU/WhileBreak>



Օրինակ

```
#include <iostream>
int main() {
    int i = 0;
    while (i <= 10) {
        std::cout << i << std::endl;
        i = i + 1;
        if (i > 5) {
            break;
        }
    }
    std::cout << "Loop end";
}
```

break հրամանից
հետո կատարվում է
այս հրամանը

0
1
2
3
4
5
Loop end

<https://repl.it/@SevakRAU/WhileBreak>



Օրինակ. break անվերջ ցիկլում

```
#include <iostream>
int main() {
    int i = 0;
    while (1) {
        std::cout << i << std::endl;
        i = i + 1;
        if (i == 5) {
            break;
        }
    }
    std::cout << "Loop end";
}
```

Անվերջ ցիկլ



0
1
2
3
4
Loop end



Օրինակ. break անվերջ ցիկլում

```
#include <iostream>
int main() {
    int i = 0;
    while (1) {
        std::cout << i << std::endl;
        i = i + 1;
        if (i == 5) {
            break;
        }
    }
    std::cout << "Loop end";
}
```

Ցիկլը
ավարտելու
պայման



0
1
2
3
4
Loop end



Օրինակ. break անվերջ ցիկլում

```
#include <iostream>
int main() {
    int i = 0;
    while (1) {
        std::cout << i << std::endl;
        i = i + 1;
        if (i == 5) {
            break;
        }
    }
    std::cout << "Loop end";
}
```

break հրամանից
հետո կատարվում է
այս հրամանը

0
1
2
3
4
Loop end

Խնդիրներ

1. Անվերջ ցիկլում մուտքագրել ամբողջ թվեր, ավարտել ցիկլը եթե մուտքագրված թիվը զույգ/կենստ է:
2. Անվերջ ցիկլում մուտքագրել ամբողջ թվեր, ավարտել ցիկլը եթե ընթացիկ մուտքագրված թիվը ավելի փոքր է քան նախորդ մուտքագրվածը:



continue հրաման

- **continue** հրամանը թույլ է տալիս բաց թողել ցիկլի ընթացիկ իտերացիան

```
#include <iostream>
```

```
int main() {  
    int i = 0;  
    while (i < 10) {  
        if (i == 5) {  
            i += 1;  
            continue;  
        }  
        std::cout << i << std::endl;  
        i += 1;  
    }  
}
```



0
1
2
3
4
6
7
8
9

5 չի տպվել

<https://repl.it/@SevakRAU/WhileContinue>



continue հրաման

- **continue** հրամանը թույլ է տալիս բաց թողել ցիկլի ընթացիկ իտերացիան

```
#include <iostream>
```

```
int main() {  
    int i = 0;  
    while (i < 10) {  
        if (i == 5) {  
            i += 1;  
            continue;  
        }  
        std::cout << i << std::endl;  
        i += 1;  
    }  
}
```

Առանց այս տողի
կունենանք անվերջ
ցիկլ

0
1
2
3
4
6
7
8
9



<https://repl.it/@SevakRAU/WhileContinue>



Խնդիրներ

1. Տպել $[1, 100]$ միջակայքի բոլոր ամբողջ թվերը, բացի նրանցից որոնք 10 վրա բաժանվում են անմնացորդ:



do while ցիկլ

Ցիկլի մարմինը առնվազն 1 անգամ կկատարվի, իսկ հետո քանի դեռ **expression**-ի արժեքը *false կամ 0* չէ կշարունակի կատարել **ցիկլի մարմինը** :

```
do {  
    statement1;  
    statement2;  
    ....  
} while(expression);
```

```
#include <iostream>  
  
int main() {  
    do {  
        std::cout << "One time will be executed" << std::endl;  
    } while (false);  
}
```

<https://repl.it/@SevakRAU/DoWhile>



One time will be executed



do while ցիկլ, օրինակ

```
#include <iostream>
```

```
int main() {  
    int i = 0, x;  
    std::cin >> x;  
    do {  
        std::cout << i << std::endl;  
        i += 1;  
    } while (i < x);  
}
```

<https://repl.it/@SevakRAU/DoWhileEx1>



Տնային աշխատանք

← → ↻ earth.ispras.ru/rau_gov/ ☆

Վարժություններ


- 00 Նախավարժանք
- 01 Թվաբանություն և ճյուղավորում
- 02 Ցիկլեր և ստատիկ զանգվածներ
- 03 Դինամիկ զանգվածներ և ֆունկցիաներ
- 04 Երկչափ զանգվածներ և դասեր

https://earth.ispras.ru/rau_gov/



Տնային աշխատանք

1-3, 5-6 խնդիրները տնային աշխատանք (4-ից բացի)



Севак Саргсян [Մաս 2: Ցիկլեր և ստատիկ զանգվածներ]: Сдать решение

НастройкиИнфоИтогПосылкиПоложение участниковОтправить вопросСообщенияВыйти из системы [sevaksargsyan]

16:55:25 / RUNNING

1234567891011121314151617181920

Сдать решение задачи 6-Ճիշտ K հատ

Ограничение времени: 1 с

Ограничение памяти: 64M

Оставшиеся послылки: 16

Задача 6: Ճիշտ K հատ

Պահանջվում է գրել ծրագիր, որը մուտքում կստանա N և K բնական թվերը, ապա հաջորդականություն N ամբողջ տարրերից կազմված: Ծրագիրը որպես արդյունք պետք է արտադի 'YES' եթե N թվերից ճիշտ K հատը մեծ են 123 -ից և 'NO' հակառակ դեպքում: Արգելվում է օգտագործել զանգված: (դաս 5, 6)

Примеры



Շնորհակալություն. Հարցե՞ր

