

Ciklusok 1 - if, while

- vezérlési szerkezet, adott utasítások ismétlését hivatott kezelni

```
for (kezdőérték; feltétel; érték léptetése)
{
    ciklusmag
}
```

pl.:

```
for (i=1; i<11; i++)
{
    utasítások
}
```

For ciklus

```
i = 0
i = 11
```

```
In [16]: #include <stdio.h>

int main(){
    for (int i = 0; i < 11; i++){
        printf("%d ", i);
    }
    printf("\nKész!");
}
```

0 1 2 3 4 5 6 7 8 9 10
Kész!

While ciklus

```
i = 0
i = 11
```

```
In [17]: #include <stdio.h>

int main(){
    int x = 0;
    while (x < 11){
        printf("%d ", x);
        x++;
    }
    printf("\nKész!");
}
```

0 1 2 3 4 5 6 7 8 9 10
Kész!

Do.. While ciklus

```
i = 0
i = 11
```

```
In [19]: #include <stdio.h>

int main(){
    int i = 0;
    do {
        printf("%d ", i);
        i++;
    }
    while (i < 11 );
    printf("\nKész!");
}
```

0 1 2 3 4 5 6 7 8 9 10
Kész!

Példa - Egymásba ágyazott ciklusok

Írassa ki a 10*10-es szorzótábla eredményeit!

- formázza be a megadott minta alapján
- jelenítse meg a műveleteket is

Minta:

```

    1  2  3  4  5  6  7  8  9 10
1  1  2  3  4  5  6  7  8  9 10
2  2  4  6  8 10 12 14 16 18 20
3  3  6  9 12 15 18 21 24 27 30
4  4  8 12 16 20 24 28 32 36 40
5  5 10 15 20 25 30 35 40 45 50
6  6 12 18 24 30 36 42 48 54 60
7  7 14 21 28 35 42 49 56 63 70
8  8 16 24 32 40 48 56 64 72 80
9  9 18 27 36 45 54 63 72 81 90
10 10 20 30 40 50 60 70 80 90 100
Minta:
```

```

    1      2      3      4      5      6      7      8      9
10
1|1 * 1 = 1|1 * 2 = 2|1 * 3 = 3|1 * 4 = 4|1 * 5 = 5|1 * 6 = 6|1 * 7 = 7|1 * 8 = 8|1 * 9 =
9|1 *10 = 10|
2|2 * 1 = 2|2 * 2 = 4|2 * 3 = 6|2 * 4 = 8|2 * 5 = 10|2 * 6 = 12|2 * 7 = 14|2 * 8 = 16|2 * 9 =
18|2 *10 = 20|
3|3 * 1 = 3|3 * 2 = 6|3 * 3 = 9|3 * 4 = 12|3 * 5 = 15|3 * 6 = 18|3 * 7 = 21|3 * 8 = 24|3 * 9 =
27|3 *10 = 30|
4|4 * 1 = 4|4 * 2 = 8|4 * 3 = 12|4 * 4 = 16|4 * 5 = 20|4 * 6 = 24|4 * 7 = 28|4 * 8 = 32|4 * 9 =
36|4 *10 = 40|
(...)
```

```
In [2]: #include <stdio.h>

int main(){
    printf("%3c ", ' ');
    for (int i = 1; i <= 10; i++) {
        printf("%3d ", i);
    }
    printf("\n%3c ", ' ');
    for (int i = 1; i <= 10; i++) {
        printf("%3c ", '-');
    }
    for (int i = 1; i <= 10; i++){
        printf("\n%3d|", i);
        for (int j = 1; j <= 10; j++){
            printf("%3d ", i*j);
        }
    }
}
```

```

    1  2  3  4  5  6  7  8  9 10
1| -  -  -  -  -  -  -  -  -  -
2| 2  4  6  8 10 12 14 16 18 20
3| 3  6  9 12 15 18 21 24 27 30
4| 4  8 12 16 20 24 28 32 36 40
5| 5 10 15 20 25 30 35 40 45 50
6| 6 12 18 24 30 36 42 48 54 60
7| 7 14 21 28 35 42 49 56 63 70
8| 8 16 24 32 40 48 56 64 72 80
9| 9 18 27 36 45 54 63 72 81 90
10|10 20 30 40 50 60 70 80 90 100
```

1. Feladat

Készítsen programot ciklus segítségével, ami kiírja az első 10 számot!

- Módosítsa úgy a programot, hogy a felhasználó határozza meg a kiírt számsor hosszát
- A program csak a páratlan számokat írja ki a ciklus segítségével

```
In [ ]: #include <stdio.h>

int main (void)
{
    int i;

    printf("Ez a program ciklus segítségével kiírja az első 10 számot.")

    for (i=1; i<11; i++)
    {
        printf("%d ", i);
    }
    return 0;
}
```

2. Feladat

Készítsen programot, ami a felhasználótól bekér egy számot, majd visszaadja az addig található számok összegét!

- Egészítse ki a programot úgy, hogy a bevitt szám faktoriálisát kiszámolja!
- Bővítse a programot, hogy a felhasználó döntse el, milyen műveletet szeretne elvégezni (összeg, vagy faktoriális)
- Az eredmény kírítása után ajánlja fel a felhasználó számára az újbóli futtatás lehetőségét
- A program legyen képes kiírni az elvégzett műveletet is

```
In [ ]: #include <stdio.h>

int main (void)
{
    int szam, i;
    int faktorialis = 1;
    int szum = 0;
    int state;

    printf("A program visszaadja a határértékig előforduló egész számok összegét és a bevitt szám faktoriálisát
    printf("Adj meg egy pozitív egész számot: ");
    scanf("%d", &i);
    szam=i;

    printf("0 - faktoriális\n1 - összeg");
    scanf("%d \n", &state);

    if (state==0) {
        for(i; i >= 1; i--)
        {
            faktorialis *= i;
        }
        printf("%d! = %d", szam, faktorialis);
    }
    else {
        for(i; i >= 1; i--)
        {
            szum += i;
        }
        printf("A számok összege %d-ig: %d \n", szam, szum);
    }

    return 0;
}
```

3. Feladat

Készítsen programot, aminek ciklusa 1-től 100-ig megy és megvizsgálja a számokat hárommal és ötten való oszthatóság szempontjából. Ha mindegyikkel osztható maradék nélkül, akkor írja ki a számot.

- Módosítsa a kódot úgy, hogy a ciklus hosszát a felhasználó adhassa meg
- Módosítsa a kódot úgy, hogy a vizsgált intervallumot (kezdő és végérték beletartozik) a felhasználó adja meg
- Bővítse a kódot további oszthatósági ellenőrzésekkel

```
In [ ]: #include <stdio.h>

int main (void)
{
    int i;
    int hossz;

    printf("Kérem, adja meg a ciklus kezdőértékét! \n");
    scanf("%d", &i);
    printf("Kérem, adja meg a ciklus végső értékét! \n");
    scanf("%d", &hossz);

    for(i; i<=hossz; i++)
    {
        if (i%3==0)
            printf("3-al osztható \n");
        if (i%5==0)
            printf("5-el osztható \n");
        if ((i%3!=0) && (i%5!=0))
            printf("%d", i);
        printf("\n");
    }

    return 0;
}
```

4. Feladat

Készítsen programot, ami ciklus segítségével az alábbi mintát rajzolja ki!

```
*****
*****
*****
*****
****
***
**
*
```

```
In [ ]: #include <stdio.h>

int main (void){
    int i, j, star=10;

    for (i=1; i<11; i++){
        for (j=1; j<=star; j++){
            printf("**");
        }
        star--;
        printf("\n");
    }
    return 0;
}
```

5. Feladat - Háromszög

Készítsen programot, ami ciklus segítségével felhasználó által megadott (átfogó szélessége, vagy sorok száma alapján) nagyságú egyenlőszárú háromszöget rajzolja ki!

7 karakter szélességű alap esetén:

```

    *
    ***
    *****
    *****
```

6. Feladat - Pascal háromszög

Készítsen programot, ami kirajzolja Pascal háromszögét!

Minta:

input = 5

```

    1
  1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

```
In [14]: #include <stdio.h>

int main(){
    int row,empty,i,j;
    int c = 1;

    printf("Adja meg a sorok számát: ");
    scanf("%d", &row);
    for (i = 0; i < row; i++){
        for (empty = 1; empty <= row - i; empty++){
            printf("%2c", ' ');
        }
        for (j = 0; j <= i; j++){
            if (j == 0 || i == 0)
                c = 1;
            else
                c = c * (i - j + 1) / j;
            printf("%4d",c);
        }
        printf("\n");
    }
}
```

Adja meg a sorok számát:

```

    1
  1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

In []: