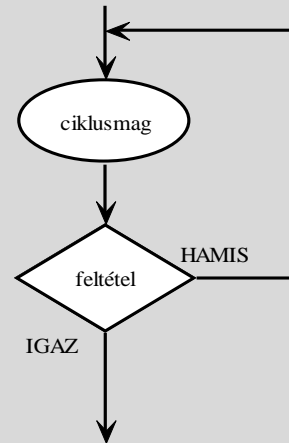


INFORMATIKA LABORATÓRIUM I.

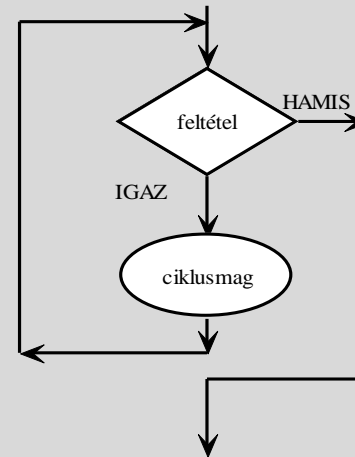
Alapok, ciklusok

Folyamatábra

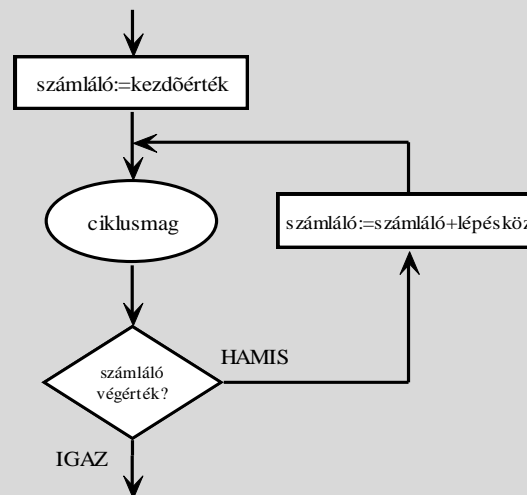
Hátultesztelő ciklus



Elöltesztelő ciklus



Számlálósos ciklus



Mondatszerű

a. számlálósos:

CIKLUS

számláló:=kezdőérték-TŐL végérték-IG lépésköz-ZEL

ciklusmag

CIKLUS VÉGE

b. hátultesztelő:

CIKLUS

ciklusmag

AMÍG feltétel

CIKLUS VÉGE

c. előltesztelő:

CIKLUS

AMÍG feltétel

ciklusmag

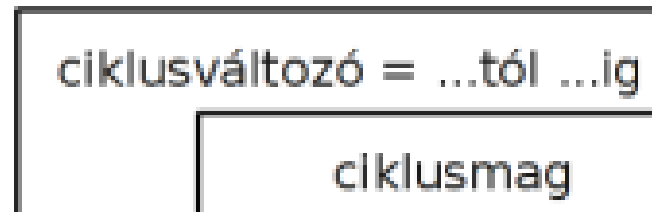
CIKLUS VÉGE

Struktogram

elől tesztelő ciklus



növekményes ciklus



hátral tesztelő ciklus



Iteráció - for utasítás

Alapvető formája a következő:

***for(inicializálás; feltétel; inkrementálás)
utasítás;***

- Az inicializálás rendszerint egy értékadó utasítás, amelyben a ciklusváltozó kap értéket.
- A feltétel egy relációs kifejezés, amely meghatározza, hogy a ciklus meddig hajtódjon végre.
- Az inkrementálás a ciklusváltozó értékét változtatja meg a ciklus minden ismétlésekor.
- A for ciklus addig hajtódik végre, amíg a feltétel igaz. Amikor a feltétel hamissá válik, akkor a program a for-t követő utasításon folytatódik.
- Nem kötelező kitölteni a fejléc elemeit, lehetőség van a hiányos for készítésére

Iteráció - for utasítás - példák

1. A következő program 1-től 100-ig megjeleníti a számokat a képernyőn.

```
main()
{
    int x;
    for ( x = 1; x <= 100; x++)
        printf ( "%d ", x );
}
```

2. Példa arra, hogy a ciklusváltozó értéke csökkenhet is.

```
for ( x = 100; x != 65; x -= 5){
    z = sqrt ( x );
    printf ( "The square root of %d, %d", x, z);
}
```

3. A feltétel kiértékelése a ciklus elején történik, így ha a feltétel a ciklus kezdetekor hamis, akkor a ciklus egyszer sem kerül végrehajtásra. Erre példa a következő:

```
x = 0 ;
for ( y = 0; y != x; ++y)
    printf ( "%d", y );
```

Iteráció - for utasítás - példák

1. Többszörös ciklusváltozó használata

```
megfordit( s, r )  
char *s,*r;  
{  
    int i, j;  
    for ( i = strlen (s) -1, j = 0; i !=0; j++, i-- )  
        r [ i ] = s [ j ] ;  
    r[ j ] = '\0';  
}
```

2. Végtelen for ciklus

```
for ( ; ; ) printf ( " Fut, fut, fut....\n" );
```

Ezt megszakíthatja a ciklus törzsén belül elhelyezett **break** utasítás a következőképpen:

```
ch= '\0';  
for ( ; ; ) {  
    ch = getchar();  
    if (x == 'A') break;  
}  
printf ( "Lenyomtad az A-t" );
```

3. for utasítás törzs nélkül

```
for ( t=0; t < SOME_VALUE; t++ ) ;
```

Időzítésre használható.

Iteráció: while utasítás

Formája:

while (feltétel) utasítás;

- ⦿ ahol az utasítás lehet
 - üres utasítás
 - egyszerű utasítás
 - egy utasításblokk.
- ⦿ A feltétel lehet egy kifejezés, és igaz bármilyen nem nulla érték.
- ⦿ A while utasítás mindaddig ismételteti a hozzá tartozó ciklustörzset, amíg az általa vizsgált kifejezés értéke igaz, vagyis nem-nulla.
- ⦿ A vizsgálat még a ciklustörzs végrehajtása előtt megtörténik.
- ⦿ Alkalmas függvények hívására, majd a további műveletek elvégzésére a ciklustörzsben. Pl: állományolvasás végjelig.

Iteráció: while utasítás példák

1. Adott karakter bekérése

```
KarakterBe()
{
    char ch;
    ch = '\0';
    while ( ch != 'q')
        ch = getchar ();
}
```

2.

```
main()
{
    char c;
    while ( ( c = getchar()) != 'q')
        putchar (c);
}
```

3. Más módon:

```
char c;
c = getchar ();
while ( c != 'q')
{
    putchar (c);
    c = getchar ();
}
```

4. for és while ciklussal ugyanaz a feladat:

```
unsigned strscn(c,s)
char s[ ];
char c;
{
    unsigned i;
    for ( i = 0; s [ i ] != c && s [ i ] != '\0'; ++i);
    return i;
}

//-----
while (s[i]!=c && s[i]!='\0')
    ++i;
return i;
```

Iteráció: do while utasítás

Formája:

```
do {  
    utasítás;  
} while (feltétel);
```

- ⦿ Az utasítástörzs addig hajtódik végre, amíg a feltétel igaz.
- ⦿ A do while utasítás a vizsgálatot a ciklus végén végzi el.

Iteráció: do while utasítás

Példa: az alábbi programok egy beolvasott egész szám számjegyeit írja ki fordított sorrendben.

```
main ()
{
    int number, digit;
    scanf ( "%d", &number );
    do {
        digit = number % 10;
        printf ( ""%d", digit );
        number /= 10;
    } while ( number != 0 )
}
```

Példa

Mit csinál ez az egyszerű mintaprogram?

```
#include <stdio.h>

main()
{
    int i;
    for(i=0; i<10;i++)
        printf("%d\n", i);
}
```

Példa

Írjon programot, amely egy 10 elemű egész típusú tömb elemeiről eldönti, hogy számtani sorozatot alkotnak-e! A tömböt a klaviatúráról történő beolvasással töltse fel, majd a megfelelő szöveges üzenetet írja ki a képernyőre!

Megoldás

```
int t[10];
int i=0,n;
printf("\nKerem a tomb elemeit!");
do{
    printf("\n%i.elem= ", i);
    scanf("%i",&t[i]);i++;
}while(t[i-1]!=0 && i<10);
if(i==10)
    n=i;
else
    n=i-1;
int d=t[1]-t[0];
i=2; //számtani sorozat?
while(t[i]-t[i-1]==d && i<n) {
    i++;
}
if(i==n)
    printf("\nSzamtani sorozat");
else
    printf("\nNem szamtani sorozat");
return 0;
```

Feladat

Készítsük el a folyamatábráját az alábbi mondatszerű leírással megadott algoritmusnak, majd programozzuk le C nyelv segítségével!

```
PROGRAM negativDarab
```

```
  db:=0;
```

```
  CIKLUS i:=1-TŐL 8-IG 1-VEL
```

```
    BE: szam
```

```
    HA szam<0 AKKOR db:=db+1
```

```
    ELÁGAZÁS VÉGE
```

```
  CIKLUS VÉGE
```

```
  KI: db, ' darab negatív szám volt!'
```

```
PROGRAM VÉGE
```

Egyszerű feladat ciklusokra

Készítsen programot, amely meghatározza egy egész szám számjegyeinek összegét!

Megoldás

```
//Készítsen programot, amely meghatározza egy egész szám  
//számjegyeinek összegét!
```

```
int szam, szjossz=0;  
printf("\nSzam= ");  
scanf("%i",&szam);  
while(szam!=0) {  
    szjossz=szjossz+szam%10;  
    szam=szam/10;  
}  
printf("\nSzamjegyek osszege= %i", szjossz);
```


Gyakorló feladatok

A pi értékét állítsuk elő a következő képletek alapján! N (elemszám) értékét kérjük be a felhasználótól! Értékeljük a sorokat, melyik közelíti meg kisebb n-re, nézzük meg közelítési időt is!

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots$$

$$\frac{\pi}{2} = \frac{2}{1} \cdot \frac{2}{3} \cdot \frac{4}{3} \cdot \frac{4}{5} \cdot \frac{6}{5} \cdot \dots$$

$$\frac{2}{\pi} = \frac{\sqrt{2}}{2} \cdot \frac{\sqrt{2+\sqrt{2}}}{2} \cdot \frac{\sqrt{2+\sqrt{2+\sqrt{2}}}}{2} \dots$$

Gyakorló feladat

Szimuláljuk egy N -szeres kockadobást. A dobások számát kérjük be a felhasználótól.

- Írassuk ki, hogy hányszor volt páratlan a dobás?
- Írassuk ki, hogy hányszor volt prím a dobás?

Gyakorló feladat - Fibonacci

Írjunk programot, amely kilistázza a képernyőre a Fibonacci számsor első n tagját. Az n értékét a program a klaviatúráról várja. A Fibonacci számsor kiszámítására a következő szabály érvényes :

Fibonacci - megoldás

```
#include <stdio.h>
#include <conio.h>

void main()
{
    int n, fee, fe, fib, i;
    printf("Kérem n értékét : ") ;
    scanf("%d",&n) ;

    fee = 0 ; fe = 1 ;
    for(i=2; i<=n; i++)
    {
        fib = fe + fee;
        printf("\nFIB[%3d] = %5d",i,fib);
        fee = fe ; fe = fib ;
    }

    printf("\n");
    getch();
}
```