

Egy [derék holland matematikus](#) szerint a [strukturált programok](#) három vezérlési szerkezetből épülnek fel: **szekvencia** (utasítások egymás utáni végrehajtása), **szelekció** (feltételes elágazás) és **iteráció** (feltételes utasítás-ismétlés). Eddigi csak szekvenciákat használtunk, most a szelekció következik.

Elágazások – szelekció

Fajtái:

egyirányú

kétirányú

többsíriányú

Egyirányú elágazás:

Algoritmus:

Ha feltétel, akkor utasítás

Megvalósítása JavaScriptben:

if (feltétel)

```
{  
    utasítás  
}
```

Döntsük el egy számról, hogy páros, vagy páratlan! A kiértékeléstől függően írjuk ki a „páros”, vagy „páratlan” szöveget. (Egy szám páros, ha kettővel osztva a maradék = 0)

```
var a = 42; // ez egy szám (number) típusú változó  
if (a % 2 == 0) // a zárójelen belül egy logikai állítás  
{ // akkor fut le, ha az állítás IGAZ  
    document.write(a, ' szám páros<br>');  
} // itt van vége a feltételnek
```

Az **if** utasítás után zárójelben egy olyan állítást kell megadnunk, amelynek logikai eredménye lesz (igaz/hamis). Az **a % 2** művelet után álló „==” jelentése: *egyenlő-e?* Ne keverjük össze az „=” jellel, amely értékadást takar. Ez egy kérdés, amely igaz, vagy hamis. A feltétel után írt „{ }” jelek közé írt utasítások akkor futnak le, ha a feltétel igaz, ellenkező esetben a fordító átugorja a blokkot

(a kapcsos zárójel közé tett részt). A változó értékét páratlanra módosítva a mintaprogram nem ír ki semmit.

Kétirányú elágazás:

Algoritmus:

Ha feltétel, akkor utasítás1

különben utasítás2

Megvalósítása JavaScriptben:

if (feltétel)

```
{  
    utasítás1  
}  
else  
{  
    utasítás2  
}
```

```
var a = 42; //szám típusú változó  
if (a % 2 == 0) // logikai állítás  
{ // akkor fut le, ha az állítás IGAZ  
    document.write(a, ' szám páros<br>');  
}  
else  
{ // akkor fut le, ha az állítás HAMIS  
    document.write(a, ' szám páratlan<br>');  
} // itt van vége a feltételnek
```

Az **else** után írt blokkban lévő utasítások akkor futnak le, ha az **if** után írt feltétel hamisnak bizonyult.

A **prompt** utasítás segítségével futási időben kérhetünk be a felhasználótól adatot.:

```
var a = prompt('Kérek egy számot','0');// változó bekérése futási időben
```

A programozók munkájának jelentős részét teszi ki a felhasználótól kapott adatok helyességének ellenőrzése. Magyarul olyan program írása, amelyet nem lehet

hibásan megadott adatokkal tönkretenni. Ha a beviteli mezőbe szám helyett stringet írunk be, akkor a program szerint ez páratlan. A helyes megoldás az lenne, ha először ellenőriznénk, hogy számot írt-e be a kedves felhasználó!

Kódunkat egészítsük ki egy újabb elágazással

```
if (isFinite(a)) // ha "a" szám típusú, akkor...
```

```
{
```

```
// ide írjuk a korábbi kódukant
```

```
}
```

Az **isFinite()** függvény logikai **igaz** értéket ad vissza, ha a paramétere szám típus.

Korlátozzuk a bekért számot egész számra (nem egész számok esetén nem sok értelme van a páros/páratlan fogalomnak).

A **Math.floor()** függvény a paraméterként kapott szám egészrészét adja vissza.

Egy szám pedig nem egyenlő az egészrészével, ha a szám nem egész (48.8 != 48).

```
if (Math.floor(a)==a) // ha a szám egész szám,
```

```
{
```

```
}
```

**Értékadó operátorok (x=10; y=5)**

Operátor	Művelet	Példa	Eredmény=
+=	x+=y	x=x+y	x=15
-=	x-=y	x=x-y	x=5
*=	x*=y	x=x*y	x=50
/=	x/=y	x=x/y	x=2
%=	x%=y	x=x%y	x=0

**Aritmetikai operátorok (x = 5;)**

Operátor	Művelet	Példa	Eredmény
+	Összeadás	x=y+2	x=7
-	Kivonás	x=y-2	x=3
*	Szorzás	x=y*2	x=10
/	Osztás	x=y/2	x=2.5
%	Modulus (osztás maradéka)	x=y%2	x=1
++	Növelés (+1)	x=++y	x=6
--	Csökkentés	x=--y	x=4

**Relációs operátorok (x = 5;)**

Operátor	Művelet	Példa
==	egyenlő	x==8 is false
===	egyenlő és azonos típus	x===5 is true
!=	nem egyenlő	x!=8 is true
>	nagyobb	x>8 is false
<	kisebb	x<8 is true
>=	nagyobb, vagy egyenlő	x>=8 is false
<=	kisebb, vagy egyenlő	x<=8 is true

**Logikai operátorok (x=6; y=3;)**

Operátor	Művelet	Példa
&&	and	(x < 10 && y > 1) is true
	or	(x==5    y==5) is false
!	not	!(x==y) is true

## Feladatok!

1. Döntsük el egy számról, hogy milyen előjelű!
2. A külső hőmérséklet ismeretében írjuk ki, hogy fagy-e odakint!
3. Egy beolvasott számról döntse el a program hogy -30 és 40 között van-e!  
Logikai és a JavaScriptben: &&
4. Két beolvasott szám közül írassuk ki a nagyobbbat!
5. Döntsük el egy bekért adatról, hogy szám-e?
6. Készítsünk scriptet, amely bekéri a víz hőmérsékletét, majd eldönti, hogy az milyen halmazállapotú. A halmazállapot lehet folyékony, gőz, vagy szilárd.
7. Írjunk scriptet, mely bekér egy számot, és eldönti, hogy osztható-e 3-mal vagy 4-gyel!
8. Adott évről döntse el a program, hogy szökőév-e! (Az év szökőév, ha az évszám osztható 4-gyel)
9. Írj olyan scriptet, ami a 16-nál kisebb számot szorozza meg 10-el, a 16-nál nagyobb vagy egyenlő számokat osztja 3-al

Az eredményeket minden esetben írasd ki!