



**BME** **KJIT**  
Budapesti Műszaki és Gazdaságtudományi Egyetem  
Közlekedés- és Járműirányítási Tanszék

# Programozás C- és Matlab nyelven

## C programozás kurzus

### BMEKOKAM603

### Folyamatvezérlés

Dr. Bécsi Tamás

3. Előadás

## 3.2. Az if-else utasítás

- *Formája:*

```
if (kifejezés)
    1.utasítás
else
    2.utasítás
```

*Ahol az else rész opcionális.*

- *Az else mindig a legközelebbi if-hez tartozik:*

```
if (n>0)
    if (a>b)
        z=a;
    else
        z=b;
```

## 3.4. A switch utasítás

```
switch (kifejezés){  
    case állandó kifejezés: utasítások  
    case állandó kifejezés: utasítások  
    .  
    .  
    default: utasítások;  
}
```

*A default ág opcionális;*

*Az egyes eseteket break utasítással törhetjük meg.*

## 3.4. switch példa

```
char c;  
switch (c){  
  case 'a': case 'e': case 'i': case 'o':  
  case 'u':  
    printf("maganhangzo"); break;  
  case ' ':  
    printf("Space"); break;  
  default: printf ("Egyik sem");  
}
```

## 3.5. Ciklusszervezés

### while utasítás

**while** (kifejezés)  
utasítás;

*Amíg a kifejezés igaz (nem 0) addig futtatja az utasítást.*

**break**; *utasítás kilép a ciklusból*

**continue**; *utasítás a következő ellenőrzésre ugrik*

## 3.5. while példa

```
char c=0;
while(++c<10)
{

printf("%d ",c);
}
```

**Eredmény: 1 2 3 4 5 6 7 8 9**

## 3.5. while példa continue

```
char c=0;
while(++c<10)
{
    if (c==3) continue;

    printf("%d ",c);
}
```

**Eredmény: 1 2 4 5 6 7 8 9**

## 3.5. while példa break

```
char c=0;
while(++c<10)
{
    if (c==3) continue;
    if (c==6) break;
    printf("%d ",c);
}
```

**Eredmény: 1 2 4 5**



## 3.5. Ciklsszervezés for utasítás

Budapesti Műszaki és Gazdaságtudományi Egyetemtem Közlekedés- és Járműirányítási Tanszék

```
for (kif1; kif2; kif3)  
    utasítás
```

*Ilyen formájában ekvivalens:*

```
kif1;  
while (kif2){  
    utasítás  
    kif3;  
}
```

*Bármelyik kifejezés elhagyható: for ( ; ; )*

## 3.5. for példa

```
for(c= 'a' ;c< 'f' ;c++)  
{  
  
    printf("%c ",c);  
  
}  
printf("Ertek ciklus utan: %c ",c);
```

**Eredmény: a b c d e Ertek ciklus utan: f**

## 3.5. for példa continue

```
for(c='a';c<'f';c++)  
{  
    if (c=='b') continue;  
  
    printf("%c ",c);  
}  
printf("Ertek ciklus utan: %c ",c);
```

**Eredmény: a c d e Ertek ciklus utan: f**

## 3.5. for példa break

```
for(c='a';c<'f';c++)
{
    if (c=='b') continue;
    if (c=='e') break;
    printf("%c ",c);
}
printf("Ertek ciklus utan: %c ",c);
```

**Eredmény: a c d Ertek ciklus utan: e**

## 3.6. Ciklusszervezés do-while utasítás

Budapesti Műszaki és Gazdaságtudományi Egyetem Közlekedés- és Járműirányítási Tanszék

```
do  
    utasítás  
while (kifejezés);
```

*Hátultesztelés (amíg a kifejezés igaz)*

## 3.6. do-while példa

```
int i=1;
do
{
    printf("%d", ++i);
} while(i<4);
```

Eredmény: 234

## 3.6. do-while példa

```
int i=1;
do
{
    printf("%d", i++);
} while(i<4);
```

Eredmény: 123

## 3.8. goto utasítás és címkék

*Példa:*

```
for (i=0;i<10;i++)  
{  
    printf("%d",i);  
    if (i==5) goto vege;  
}  
vege: printf("vege");
```



# Köszönöm a figyelmet

Budapesti Műszaki és Gazdaságtudományi Egyetem *Közlekedés- és Járműirányítási Tanszék*

