

## 5 Karakterláncok (string)

- karakterláncok (**string** típusú változók)

### 5.1 Karakterláncok (string típusú változók)

A **string** típusú változókba szöveget olvashatunk be. Az ilyen változó maximális hossza 255 karakter.

**Feladat:** Készítsünk programot, amely beolvasson egy mondatot, majd kiírja hány karakterből áll a mondat.

**Megoldás:** A feladat megoldásához bevezetünk egy *s* változót, melynek típusa **string**. Beolvasás után ennek a változónak a hosszát a **length()** függvénnyel kapjuk meg. Programunk így néz ki:

```
program Pelda16;
uses crt;
var s:string;
begin
  clrscr;
  write('Írj be egy mondatot: ');
  readln(s);
  writeln('A mondat hossza: ',length(s),' karakter. ');
  readln;
end.
```

**Feladat:** Készítsünk programot, amely beolvasson egy mondatot, majd kiírja a mondatot nagy betűkkel.

**Megoldás:** A feladat megoldásához bevezetünk egy *s* változót, melynek típusa **string**. Beolvasás után a mondat minden egyes karakterét egy cikluson belül megváltoztatjuk nagy betűre. Az *s* változóban levő mondat első betűjét az **s[1]** adja meg, második betűjét az **s[2]** adja meg, stb. Egy karaktert nagy betűre az **upcase()** függvény segítségével változtathatunk meg. Programunk így néz ki:

```
program Pelda17;
uses crt;
var s:string;
    i:integer;
begin
  clrscr;
  write('Írj be egy mondatot: ');
  readln(s);
  for i:=1 to length(s) do
    s[i]:=upcase(s[i]);
  writeln(s);
  readln;
end.
```

**Feladat:** Készítsünk programot, amely kiírja a karakterek ASCII kódját a 32-es kódtól (helyköz) a 255-ös kódig.

**Megoldás:** Ha tudjuk egy karakter ASCII kódját, a karaktert a **chr()** függvénnyel tudjuk kiírni. Programunk így néz ki:

```

program Pelda18;
uses crt;
var i:integer;
begin
  clrscr;
  for i:=32 to 255 do
    begin
      write(i:6);
      write(chr(i):2);
    end;
  readln;
end.

```

Ha egy karakternek szeretnénk megtudni, hogy mi az ASCII kódja, akkor azt az **ord()** függvénnyel határozhatjuk meg. Pl. **ord('A')** értéke 65, mivel az A betű ASCII kódja 65. Az **ord()** függvény pont az ellentettje a **chr()** függvénynek. A karakterek ASCII kódját és az említett két függvényt fogjuk kihasználni olyan programok készítésénél, ahol a programunkat a billentyűzet segítségével fogjuk irányítani (pl. nyilak segítségével játékprogramnál).

#### ASCII kódtábla (32-255):

32=	33=!	34="	35=#	36=\$	37=%	38=&	39='	40=(	41=)
42=*	43=+	44=,	45=-	46=.	47=/	48=0	49=1	50=2	51=3
52=4	53=5	54=6	55=7	56=8	57=9	58=:	59=;	60=<	61==
62=>	63=?	64=@	65=A	66=B	67=C	68=D	69=E	70=F	71=G
72=H	73=I	74=J	75=K	76=L	77=M	78=N	79=O	80=P	81=Q
82=R	83=S	84=T	85=U	86=V	87=W	88=X	89=Y	90=Z	91=[
92=\	93=]	94=^	95=_	96=`	97=a	98=b	99=c	100=d	101=e
102=f	103=g	104=h	105=i	106=j	107=k	108=l	109=m	110=n	111=o
112=p	113=q	114=r	115=s	116=t	117=u	118=v	119=w	120=x	121=y
122=z	123={	124=	125=}	126=~	127=•	128=€	129=□	130=,	131=f
132=„	133=...	134=†	135=‡	136=^	137=‰	138=Š	139=◀	140=œ	141=□
142=Ž	143=□	144=□	145='	146='	147="	148="	149=•	150=–	151=—
152=˜	153=™	154=š	155=›	156=œ	157=□	158=ž	159=Ÿ	160=	161=ı
162=¢	163=£	164=¤	165=¥	166=ı	167=§	168="	169=©	170=ª	171=«
172=¬	173=	174=®	175=¯	176=°	177=±	178=²	179=³	180=´	181=µ
182=¶	183=·	184=¸	185='	186=º	187=»	188=¼	189=½	190=¾	191=¿
192=À	193=Á	194=Â	195=Ã	196=Ä	197=Å	198=Æ	199=Ç	200=È	201=É
202=Ê	203=Ë	204=Ì	205=Í	206=Î	207=Ï	208=Ð	209=Ñ	210=Ò	211=Ó
212=Ô	213=Õ	214=Ö	215=×	216=Ø	217=Ù	218=Ú	219=Û	220=Ü	221=Ý
222=Þ	223=ß	224=à	225=á	226=â	227=ã	228=ä	229=å	230=æ	231=ç
232=è	233=é	234=ê	235=ë	236=ì	237=í	238=î	239=ï	240=ð	241=ñ
242=ò	243=ó	244=ô	245=õ	246=ö	247=÷	248=ø	249=ù	250=ú	251=û
252=ü	253=ý	254=þ	255=ÿ						

## Feladatok:

1. Készítsünk programot, amely bekér egy mondatot, majd kiírja ugyanezt a mondatot úgy, hogy mindegyik betű (karakter) után kirak egy szóközt. **(string01)**

```
Ird be a mondatot: Ez egy szep
nap.
E z   e g y   s z e p   n a p .
```

2. Kérjünk be egy keresztnévet, majd írassuk ki ezt a nevet betűnként függőlegesen lefelé a képernyőre. **(string02)**  
Például ha megadjuk névnek a "Peti"-t, a program írja ki ezt:

```
P
e
t
i
```

```
Ird be a nevet: Zsofia
Z
s
o
f
i
a
```

3. Készítsünk programot, amely bekér egy mondatot, majd kiírja ugyanezt a mondatot fordítva. **(string03)**  
Például ha beírjuk "Szep napunk van ma.", kiírja ezt: ".am nav knupan pezS"

```
Ird be a mondatot: Vagyok aki
vagyok.
.koygav ika koygaV
```

4. Olvassunk be egy **A** természetes számot és egy **CH** karaktert (**char** típusú változót - hasonlóan mint a **string** típust, csak a deklarálásnál **string** helyett **char**-t írunk). Rajzoljunk ki a beolvasott karakterből egy **A** oldalú négyzetet a képernyőre (minden sorban **A** drb. karakter legyen és összesen **A** drb. sorunk legyen) egymásba ágyazott cilusok segítségével. **(string04)**  
Például **A=4**-re és **CH='M'**-re rajzolja ki a program ezt:

```
MMMM
MMMM
MMMM
MMMM
```

```
Ird be a betut: A
Ird be a szamot: 6
AAAAAA
AAAAAA
AAAAAA
AAAAAA
AAAAAA
```

5. Kérjünk be egy keresztnévet, és egy db számot, majd írassuk ki ezt a nevet betűnként függőlegesen lefelé a képernyőre, soronként annyiszor a betűket amennyi db számot megadtunk **(string05)**  
Például ha megadjuk névnek a "Peti"-t, és a **db-r 4** et a program írja ki ezt:

```
PPPP
eeee
tttt
iiii
```

```
Ird be a nevet: Peti
Hanszor: 6
PPPPPP
eeeeee
tttttt
iiiiiii
```

6. Kérjél be egy 6 karakterből álló keresztnévet! Az név első felét, és a második felét külön sorba írasd ki! (pl: Zsófia) **(string06)**

```
Ird be a 6 karakter hosszú nevet:
Zsofia
_____
Zso
fia
```