

## Összetett adattípusok

- **Egyszerű** adattípusok: egy változóban **egy** adat.
- **Összetett** adattípusok: egy változóban **több** adat.
- Megoldások **szubrutinnal**:
  - Feltesszük, hogy az **input** adatok már rendelkezésre állnak, ezeket megkapjuk, míg az **output** eredményeket visszaadjuk.
  - A szubrutin neve mögött zárójelben felsoroljuk a kommunikációt végző **paramétereket**, azaz az input ill. output adatokat tartalmazó változók azonosítóit.
  - A megoldás csak az **érdemi részt** tartalmazza, az esetleges adatbekérés és eredménykiírás, a szubrutint hívó programrész (pl. a főprogram) feladata.



## Tömbök

- A **tömb** egy általánosan és gyakran használt eszköz a szoftverfejlesztők, programozók körében, mivel a tömbökkel kényelmesen kezelhetünk **több, azonos típusú** adatot.
- Az adatok elérésére, a tömbök egy elemének kiválasztására sorszámokat, más néven **indexeket** használunk.
  - Egy index: **egydimenziós** tömb.
  - Két index: **kétdimenziós** tömb.
  - ...
  - $N$  index:  **$N$  dimenziós** tömb.



# Tömbök

## ■ Tömbadat

- Egydimenziós tömb esetén egy adatsornak, vagy matematikai fogalommal egy vektornak, kétdimenziós tömb esetén egy adattáblázatnak vagy mátrixnak felel meg.

## ■ Tömb adattípus

- Definiálja a tömb dimenzióit és a tömb elemeinek típusát.

## ■ Tömbváltozó

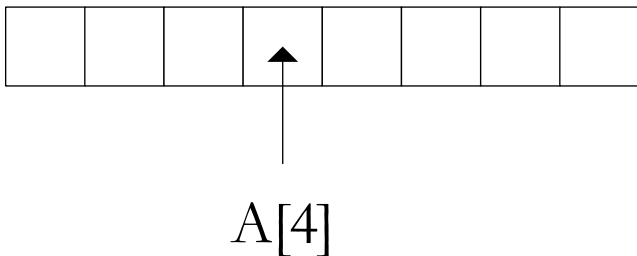
- Ebben tárolódnak az adatok. Egy elem kiválasztásával, azaz a tömbváltozó indexelésével egy, a tömb elemtípusával megegyező típusú változót hivatkozunk.

## ■ Megjegyzés:

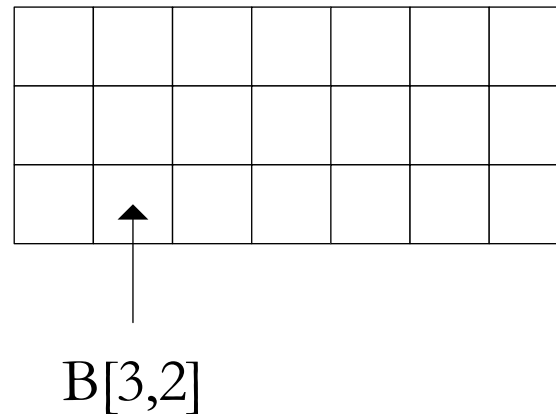
- A tömbök **indexelésére** 1-től kezdődő sorszámokat használunk, azaz minden dimenzió legkisebb indexe 1.
- A tömbök **elemtípusa** tetszőleges egyszerű ill. összetett adattípus lehet.

# Tömbök

Az  $A[8]$  egydimenziós tömb



A  $B[3,7]$  kétdimenziós tömb



**Tömbök és elemeik**

# Tömbök

## ■ Deklarálás

- Definiálnunk kell a tömb elemeinek típusát és a tömb dimenzióit.

## ■ Konstans

SORMAX 10  
OSZLMAX 20

## ■ Típus

MATRIX Kétdimenziós egész tömb[SORMAX, OSZLMAX]

Funkció	Azonosító	Típus	Jelleg
Egy adatsor	A	Egydimenziós valós tömb	I
Az adatsor adatainak száma	N	Egész	I
Egy mátrix	B	MATRIX	I, M, O
A mátrix sorainak száma	S	Egész	I
A mátrix oszlopainak száma	O	Egész	I

# Tömbök

## ■ Hivatkozás

- A tömbök elemeire a tömbváltozó azonosítója után, szögletes zárójelbe tett, egymástól vesszővel elválasztott indexekkel hivatkozunk. Az **indexek** tetszőleges **egész kifejezések** lehetnek.

Pl.	A[1]	/* Az A tömb első eleme */
	A[I+1]	/* Az A tömb I+1. eleme */
	B[S,2]	/* A B tömb S. sorának 2. eleme */

## ■ Tömbkonstans

Pl.	SZINEK=("Treff", "Káró", "Kőr", "Pikk")
	MTX=((1,2,3),(4,5,6))

## ■ Megjegyzés:

- A tömbökre nem definiálunk műveleteket.
- A tömbelemekre minden olyan művelet megengedett, amely a tömb elemtípusára definiált.



## Átlagnál nagyobb elemek

- **Feladat:** Adott  $N$  db szám, mint input adat. Kérjük be őket és írjuk ki az átlagnál nagyobb adatokat! Az adatok száma is input adat, értéke legfeljebb 100.

## Átlagnál nagyobb elemek

- **Feladat:** Adott  $N$  db szám, mint input adat. Kérjük be őket és írjuk ki az átlagnál nagyobb adatokat! Az adatok száma is input adat, értéke legfeljebb 100.

Funkció	Azonosító	Típus	Jelleg
Az adatok száma	N	Egész	I
Az adatok	A	Egydimenziós valós tömb[100]	I
Az adatok összege	OSSZ	Valós	M
Az adatok átlaga	ATL	Valós	M
Ciklusváltozó	I	Egész	M



## Átlagnál nagyobb elemek

```
/* Átlagnál nagyobb elemek */  
/* Adatbekérés */  
Be: N  
for I  $\leftarrow$  1,N  
    Be: A[I]  
/* Átlagszámolás */  
OSSZ  $\leftarrow$  0  
for I  $\leftarrow$  1,N  
    OSSZ  $\leftarrow$  OSSZ+A[I]  
ATL  $\leftarrow$  OSSZ/N  
/* Eredménykiírás */  
for I  $\leftarrow$  1,N  
    if A[I]>ATL  
        Ki: A[I]
```

## Kockadobások gyakorisága

- **Feladat:** Input adatként adott egy 6 oldalú, szabályos dobókockával dobott számsorozat. A dobások számát nem ismerjük, ez tetszőlegesen nagy lehet, a sorozat végét a 0 szám (mint végjel) jelzi. Kérjük be az adatokat és készítsünk gyakorisági táblázatot az egyes dobások előfordulási darabszámáról!



## Kockadobások gyakorisága

- **Feladat:** Input adatként adott egy 6 oldalú, szabályos dobókockával dobott számsorozat. A dobások számát nem ismerjük, ez tetszőlegesen nagy lehet, a sorozat végét a 0 szám (mint végjel) jelzi. Kérjük be az adatokat és készítsünk gyakorisági táblázatot az egyes dobások előfordulási darabszámáról!

- **Konstans**

N 6

Funkció	Azonosító	Típus	Jelleg
Az egyes dobások darabszáma	DB	Egydimenziós egész tömb[N]	M, O
Az aktuális dobás	A	Egész	I
Segédváltozó	I	Egész	M

## Kockadobások gyakorisága

```
/* Kockadobások gyakorisága */
Ki: "Kockadobások (1-",N,") gyakorisága (Kilépés:0)"
/* Kezdőértékek */
for I  $\leftarrow$  1,N
    DB[I]  $\leftarrow$  0
I  $\leftarrow$  0
repeat
    /* Bekérés */
    I  $\leftarrow$  I+1
    Ki: I, ". dobás:"
    Be: A
    if A  $\neq$  0
        DB[A]  $\leftarrow$  DB[A]+1
until A=0
Ki: "Gyakoriságok"
for I  $\leftarrow$  1,N
    Ki: I,DB[I]
```



## Érték törlése adatsorból

- **Feladat:** Töröljünk ki egy  $N$  elemű adatsorból egy adott értéket, ahányszor csak előfordul!



## Érték törlése adatsorból

- **Feladat:** Töröljünk ki egy  $N$  elemű adatsorból egy adott értéket, ahányszor csak előfordul!

Funkció	Azonosító	Típus	Jelleg
Az adatsor	A	Egydimenziós, tetszőleges elemtípusú tömb	I, M, O
Az adatok száma	N	Egész	I, O
A törlendő érték	X	Az A tömb elemeivel megegyező típusú	I
Az aktuálisan vizsgált tömbelem indexe	I	Egész	M
Az előremásolt elemek száma	J	Egész	M

## Érték törlése adatsorból

```
/* Érték törlése adatsorból */  
TOROL(A,N,X)  
/* Kezdőérték */  
J  $\leftarrow$  0  
/* Előremásolás */  
for I  $\leftarrow$  1,N  
    if A[I]  $\neq$  X  
        J  $\leftarrow$  J+1  
        A[J]  $\leftarrow$  A[I]  
/* Darabszám */  
N  $\leftarrow$  J
```



## Eratoszthenész szitája

- Prímszámok előállítására (időszámításunk előtt kb. 200 évvel):
  - Írjuk fel az 1-nél nagyobb természetes számokat egy tetszőlegesen nagy  $K$  korlátig.
  - A legkisebb szám 2, jelöljük meg és töröljük a 2-vel oszthatókat a 2 kivételével.
  - A megmaradó, jelöletlen számok közül a legkisebb 3, ezzel végezzük el ugyanezt az eljárást: jelöljük meg és töröljük a 3-mal oszthatókat a 3 kivételével.
  - A megmaradó, jelöletlen számok közül a legkisebbel mindig elvégezve az eljárást, előbb-utóbb „elfogynak” a számok, vagy töröljük őket, vagy megjelöljük.
  - A megjelölt, a szitán mintegy „fennmaradó” számok éppen a  $K$ -nál nem nagyobb prímszámok.



## Eratosztenesz szitája

- **Feladat:** Készítsünk megoldást az előző algoritmus megvalósítására tömb segítségével!



## Eratosztenesz szitája

- **Feladat:** Készítsünk megoldást az előző algoritmus megvalósítására tömb segítségével!

- **Konstans**

K 1000

/\* Eddig határozzuk meg a prímszámokat \*/

Funkció	Azonosító	Típus	Jelleg
A felírt számokat reprezentáló tömb	A	Egydimenziós logikai tömb[K]	M, O
Az aktuálisan vizsgált szám	P	Egész	M
A többszörösök törléséhez	I	Egész	M

## Eratosztesz szitája

```
/* Eratosztesz szitája tömbbel */  
/* Számok felírása */  
for I  $\leftarrow$  2,K  
    A[I]  $\leftarrow$  igaz  
/* Szitálás */  
for P  $\leftarrow$  2,K  
    /* A P szám prímszám? */  
    if A[P]  
        /* Igen, töröljük a többszöröseit */  
        for I  $\leftarrow$  2*P,K,P  
            A[I]  $\leftarrow$  hamis  
/* Kiírás */  
for I  $\leftarrow$  2,K  
    if A[I]  
        Ki: I
```



## Mátrixösszegek

- **Feladat:** Adott egy  $N \times M$ -es számmátrix. Határozzuk meg a mátrix sorainak, oszlopainak, valamint az összes elemének az összegét!

## Mátrixösszegek

- **Feladat:** Adott egy  $N \times M$ -es számmátrix. Határozzuk meg a mátrix sorainak, oszlopainak, valamint az összes elemének az összegét!

- **Konstans**

```
NMAX 10      /* Sorok maximális száma */
MMAX 10      /* Oszlopok maximális száma */
```

Funkció	Azonosító	Típus	Jelleg
A feldolgozandó mátrix	A	Kétdimenziós valós tömb[NMAX,MMAX]	I
A mátrix sorainak száma	N	Egész	I
A mátrix oszlopainak száma	M	Egész	I
A sorösszegek	SOR	Egydimenziós valós tömb[NMAX]	M, O
Az oszlopösszegek	OSZL	Egydimenziós valós tömb[MMAX]	M, O
A teljes mátrix összeg	OSSZ	Valós	M, O
Segédváltozók	I, J	Egész	M

## Mátrixösszegek

```
/* Mátrixösszegek */
OSSZEK(A,N,M,SOR,OSZL,OSSZ)
/* Kezdőértékek */
for J  $\leftarrow$  1,M
    OSZL[J]  $\leftarrow$  0
OSSZ  $\leftarrow$  0
/* Összegzés */
for I  $\leftarrow$  1,N
    SOR[I]  $\leftarrow$  0
    for J  $\leftarrow$  1,M
        SOR[I]  $\leftarrow$  SOR[I]+A[I,J]
        OSZL[J]  $\leftarrow$  OSZL[J]+A[I,J]
    OSSZ  $\leftarrow$  OSSZ+SOR[I]
```

## Oszlopok törlése mátrixból

- **Feladat:** Adott egy  $N \times M$ -es mátrix. Töröljük ki a mátrix azon oszlopait, amelyekben minden elem azonos!



## Oszlopok törlése mátrixból

- **Feladat:** Adott egy  $N \times M$ -es mátrix. Töröljük ki a mátrix azon oszlopait, amelyekben minden elem azonos!

Funkció	Azonosító	Típus	Jelleg
A feldolgozandó mátrix	A	Kétdimenziós tetszőleges elemtípusú tömb	I, O
A mátrix sorainak száma	N	Egész	I
A mátrix oszlopainak száma	M	Egész	I, O
Az aktuálisan vizsgált oszlop indexe	J	Egész	M
A vizsgált oszlop elemei egyformák-e	EGYF	Logikai	M
Az előremásolandó oszlop indexe	K	Egész	M
Segédváltozó	I	Egész	M



## Oszlopok törlése mátrixból

```
/* Egyforma elemű oszlopok törlése */
TOROL(A,N,M)
J ← 1
while J ≤ M
    /* A J. oszlop vizsgálata */
    EGYF ← igaz
    I ← 2
    while (I ≤ N) AND EGYF
        EGYF ← A[I,J] = A[1,J]
        I ← I + 1
    if EGYF
        /* A J. oszlop törlése */
        for K ← J + 1, M
            for I ← 1, N
                A[I, K - 1] ← A[I, K]
        M ← M - 1
    else
        J ← J + 1
```



## Sztring típus

Szöveges adatok használatát megengedő adattípus.

- Konstans:

pl. "Alma"

- Műveletek:

- Összefűzés (+)

- Hasonlítások:

- Mint az egészeknél, csak itt nem a számok értéke, hanem a sztringek karakterei alapján történik a hasonlítás.

Pl. "alma" + "fa" → "almafa"

"Alma" < "alma" → igaz

"Kovács" < "Kovácsné" → igaz

"Kovacs" < "Kovács" → igaz

# Függvények

## ■ Sztringkezelő függvények:

- LENGTH(X) Hossz meghatározása
- COPY(X, Y, Z) Részsztring képzése
- POS(X, Y) Részsztring keresése

## ■ Konverziós függvények:

- VAL(X) Sztring számmá alakítása
- STR(X) Szám sztringgé alakítása

Pl.    LENGTH("Maci") → 4  
      COPY("Mazsola", 3, 2) → "zs"  
      COPY("Hahó", 5, 1) → ""  
      COPY("valami", 5, 4) → "mi"  
      POS("a", "alma") → 1  
      POS("A", "alma") → 0  
      VAL("3.14") → 3.14  
      STR(3.14) → "3.14"



## Sztringek, mint karaktertömbök

- A sztringek használhatók olyan **egydimenziós tömbként** is, amelynek elemei **karakterek**.
  - Az első elem a sztring első karaktere, a második elem a második karakter és így tovább.
  - Ha ST egy sztring típusú változó, akkor az ST[1] hivatkozás egy olyan **karakteres változó**, amely a sztring első karakterét tárolja.
  - A sztring egyes karakterei így **elérhetők** és **módosíthatók** is.
  - Egy sztring karaktertömbként történő módosítása nem változtatja meg a **sztring hosszát**.



## Sztring megfordítás

- **Feladat:** Fordítsunk meg egy adott sztringet!

Pl. "Réti pipitér" → "rétipip itéR"



## Sztring megfordítás

- **Feladat:** Fordítsunk meg egy adott sztringet!

Pl. "Réti pipitér" → "rétipip itéR"

Funkció	Azonosító	Típus	Jelleg
A megfordítandó sztring	S	Sztring	I
Az eredménysztring	ER	Sztring	M, O
Ciklusváltozó	I	Egész	M

## Sztring megfordítás

/\* Sztring megfordítás 1 \*/

FORDIT1(S,ER)

ER  $\leftarrow$  ""

**for** I  $\leftarrow$  LENGTH(S),1,-1

ER  $\leftarrow$  ER+S[I]

/\* Sztring megfordítás 2 \*/

FORDIT2(S,ER)

ER  $\leftarrow$  ""

**for** I  $\leftarrow$  1,LENGTH(S)

ER  $\leftarrow$  S[I]+ER

## Sztring megfordítás

Funkció	Azonosító	Típus	Jelleg
A megfordítandó sztring	S	Sztring	I, O
A sztring hossza	H	Egész	M
Két karakter cseréjéhez	CS	Karakter	M
Ciklusváltozó	I	Egész	M

/\* Sztring megfordítás 3 \*/

FORDIT3(S)

H ← LENGTH(S)

**for** I ← 1, H DIV 2

CS ← S[I]

S[I] ← S[H+1-I]

S[H+1-I] ← CS





## Egyszerű kifejezés kiértékelése

- **Feladat:** Adott egy sztring, amelyben egy egyszerű kifejezés található. Helyes a kifejezés, ha:
  - páratlan számú karakterből áll,
  - a páratlan karakterpozíciókban számjegyek,
  - a párosokban a '+' vagy '-' műveleti jelek szerepelnek.

Döntsük el a kifejezésről, hogy helyes-e vagy sem és ha helyes, akkor számítsuk ki a kifejezés értékét!

Pl.    "1+3-2" → helyes, 2  
      "2\*4" → helytelen  
      "5" → helyes, 5  
      "+5" → helytelen



## Egyszerű kifejezés kiértékelése

Funkció	Azonosító	Típus	Jelleg
A feldolgozandó sztring	S	Sztring	I
A sztring helyessége	JO	Logikai	M, O
A kifejezés értéke	ER	Egész	M, O
A sztring hossza	H	Egész	M
Segédváltozó	I	Egész	M

## Egyszerű kifejezés kiértékelése

```
/* Egyszerű kifejezés kiértékelése */
ERTEKEL(S,JO,ER)
/* Hosszellenőrzés */
H ← LENGTH(S)
JO ← H MOD 2=1
if JO
    /* Karakterek ellenőrzése */
    I ← 1
    while (I<=H) AND JO
        if (I MOD 2=1) AND ((S[I]<'0') OR (S[I]>'9')) OR
            (I MOD 2=0) AND (S[I]<>'+' ) AND (S[I]<>'-' )
            JO ← hamis
        else
            I ← I+1
    if JO
        /* A kifejezés értékének kiszámítása */
        ER ← ASC(S[1])–ASC('0')
        for I ← 3,H,2
            if S[I–1]='+'
                ER ← ER+ASC(S[I])–ASC('0')
            else
                ER ← ER–(ASC(S[I])–ASC('0'))
```

