Algoritmusok

Jelölések:

x, y, z: a sorozatok elemeit tartalmazó tömbök
az indexelés 1-től kezdődik
n: az x sorozat elemszáma
darab: általában az eredményként kapott tömb elemszáma
T: a sorozat elemein értelmezett tulajdonság
egy elem esetén vagy igaz, vagy hamis

A logikai műveletek kiértékelésénél felhasználtuk a logikai műveletek rövidzárját.
Az esetleges megjegyzéseket aposztrófjel (’) vezeti be a mondatszerű leírásban.
Az emelt szintű érettségi követelményeiben szereplő algoritmusokat *-gal jelöltük.

Elemi algoritmusok

Sorozathoz egyetlen értéket rendelnek.

Összegezés (*)

A sorozat elemeinek összege.

```
összeg=0
ciklus i=1-től n-ig
    összeg=összeg+x(i)
ciklus vége
```

Eldöntés (*)

Van-e T tulajdonságú elem a sorozatban?

```
i=1
ciklus amíg i≤n és x(i) nem T tulajdonságú
    i=i+1
ciklus vége
van=(i≤n)
```

Eldöntés (*)

Minden elem T tulajdonságú-e a sorozatban?

```
i=1
ciklus amíg i≤n és x(i) T tulajdonságú
    i=i+1
ciklus vége
mind=(i>n)
```

Kiválasztás (*)

T tulajdonságú elem keresése a sorozatban
(tudjuk, hogy van).

```
i=1
ciklus amíg x(i) nem T tulajdonságú
    i=i+1
ciklus vége
sorszám=i
```

Lineáris keresés (*)

T tulajdonságú elem keresése a sorozatban
(nem biztos, hogy van).

```
i=1
ciklus amíg i≤n és x(i) nem T tulajdonságú
    i=i+1
ciklus vége
van=(i≤n)
ha van akkor
    sorszám=i
elágazás vége
```

Keresés rendezett sorozatban (*)

Az y értéket keressük a rendezett x sorozatban.

```
i=1
ciklus amíg i≤n és x(i)<y
    i=i+1
ciklus vége
van=(i≤n és x(i)=y)
ha van akkor
    melyik=i
elágazás vége
```

Logaritmikus (bináris) keresés (*)

Az y értéket keressük a rendezett x sorozatban.

```
első=1
utolsó=n
ciklus
    középső=(első+utolsó) div 2
    elágazás
        y<x(középső) esetén utolsó=középső-1
        y>x(középső) esetén első=középső+1
    elágazás vége
amíg első≤utolsó és x(középső)≠y
ciklus vége
van=(első≤utolsó)
ha van akkor
    melyik=középső
elágazás vége
```

Megszámlálás (*)

Hány T tulajdonságú elem van a sorozatban?

```
darab=0
ciklus i=1-től n-ig
    ha x(i) T tulajdonságú akkor
        darab=darab+1
    elágazás vége
ciklus vége
```

Maximumkiválasztás (*)

A sorozat elemeinek maximuma.

```
melyik=1
max=x(1)
ciklus i=2-től n-ig
    ha max<x(i) akkor
        melyik=i
        max=x(i)
    elágazás vége
ciklus vége
```

Összetett algoritmusok

Sorozathoz/sorozatokhoz sorozatot/sorozatokat rendelnek.

Kiválogatás kigyűjtéssel (*)

Az x tömb T tulajdonságú elemeinek kiválogatása.
az y tömbbe.

```
darab=0
ciklus i=1-től n-ig
    ha x(i) T tulajdonságú akkor
        darab=darab+1
        y(darab)=x(i)
        ’ vagy y(darab)=i ha a sorszámokat gyűjtjük ki
    elágazás vége
ciklus vége
```

Kiválogatás helyben (*)

A sorozat T tulajdonságú elemeinek kiválogatása.
az eredeti tömbbe.

```
darab=0
ciklus i=1-től n-ig
    ha x(i) T tulajdonságú akkor
        darab=darab+1
        x(darab)=x(i)
    elágazás vége
ciklus vége
```

Szétválogatás két új tömbbe

Az egyik tömbbe a T tulajdonságú elemek kerülnek,
a másikba pedig a többi elem (egymást kizáró tulajdonságok).

```
dby=0
dbz=0
ciklus i=1-től n-ig
    ha x(i) T tulajdonságú akkor
        dby=dby+1
        y(dby)=x(i) ’ vagy y(dby)=i
    egyébként
        dbz=dbz+1
        z(dbz)=x(i) ’ vagy z(dbz)=i
    elágazás vége
ciklus vége
```

Szétválogatás egy új tömbbe

A tömb elejére a T tulajdonságú elemek kerülnek,
a végére pedig a többi elem (egymást kizáró tulajdonságok).

```
darab=0
elsőnemjó=n+1
ciklus i=1-től n-ig
    ha x(i) T tulajdonságú akkor
        darab=darab+1
        y(darab)=x(i)
    egyébként
        elsőnemjó=elsőnemjó-1
        y(elsőnemjó)=x(i)
    elágazás vége
ciklus vége
```

Szétválogatás helyben

Az eredeti tömb elejére a T tulajdonságú elemek kerülnek,
a végére pedig a többi elem.

```
első=1
utolsó=n
temp=x(1)
ciklus amíg első<utolsó
    ciklus amíg  első<utolsó és
                 x(utolsó) nem T tulajdonságú
        utolsó=utolsó-1
    ciklus vége
    ha első<utolsó akkor
        x(első)=x(utolsó)
        első=első+1
        ciklus amíg  első<utolsó és
                     x(első) T tulajdonságú
            első=első+1
        ciklus vége
        ha első<utolsó akkor
            x(utolsó)=x(első)
            utolsó=utolsó-1
        elágazás vége
    elágazás vége
ciklus vége
x(első)=temp
ha x(első) T tulajdonságú akkor
    darab=első
egyébként
    darab=első-1
elágazás vége
```

Metszet

Két sorozat közös elemeinek meghatározása (feltételezzük,
hogy egy sorozaton belül egy elem csak egyszer szerepel).

```
darab=0
ciklus i=1-től elemszám(x)-ig
    j=1
    ciklus amíg j≤elemszám(y) és x(i)≠y(j)
        j=j+1
    ciklus vége
    ha j≤elemszám(y) akkor
        darab=darab+1
        z(darab)=x(i)
    elágazás vége
ciklus vége
```

Unió

Két sorozatból újabb sorozat készítése, amely a közös
elemeket csak egyszer tartalmazza (feltételezzük, hogy az
eredeti sorozatokon belül egy elem csak egyszer szerepel).

```
x elemeinek átmásolása z-be
darab=elemszám(x)
ciklus j=1-től elemszám(y)-ig
    i=1
    ciklus amíg i≤elemszám(x) és x(i)≠y(j)
        i=i+1
    ciklus vége
    ha i>elemszám(x) akkor
        darab=darab+1
        z(darab)=y(j)
    eljárás vége
ciklus vége
```

Halmazfelsorolás készítése

Az y sorozathoz hozzávesszük az x sorozat azon elemeit, amelyek még nem szerepelnek az y-ban.

```
darab=0
ciklus i=1-től n-ig
    j=1
    ciklus amíg j≤darab és x(i)≠y(j)
        j=j+1
    ciklus vége
    ha j>darab akkor
        darab=darab+1
        y(darab)=x(i)
    elágazás vége
ciklus vége
```

Összefuttatás

Rendezett sorozatokból (x ,y) egy újabb sorozat (z) készítése, amelybe a közös elemeket csak egyszer vesszük fel.

```
i=1
j=1
darab=0
ciklus amíg i≤elemszám(x) és j≤elemszám(y)
    darab=darab+1
    elágazás
        x(i)<y(j) esetén z(darab)=x(i) : i=i+1
        x(i)=y(j) esetén z(darab)=x(i) : i=i+1 : j=j+1
        x(i)>y(j) esetén z(darab)=y(j) : j=j+1
    elágazás vége
ciklus vége
ciklus amíg i≤elemszám(x)
    darab=darab+1
    z(darab)=x(i)
    i=i+1
ciklus vége
ciklus amíg j≤elemszám(y)
    darab=darab+1
    z(darab)=y(j)
    j=j+1
ciklus vége
```

Összefuttatás ütközővel

Rendezett sorozatokból (x ,y) egy újabb sorozat (z) készítése, amelybe a közös elemeket csak egyszer vesszük fel. Az eredeti sorozatokat ki tudjuk egészíteni egy újabb elemmel, amely nagyobb, mint a sorozatokban előforduló értékek (ütköző). Az x sorozat eredeti elemszáma: n, az y sorozat eredeti elemszáma: m

```
i=1
j=1
darab=0
x(n+1)=ütköző
y(m+1)=ütköző
ciklus amíg i<n+1 vagy j<m+1
    darab=darab+1
    elágazás
        x(i)<y(j) esetén z(darab)=x(i) : i=i+1
        x(i)=y(j) esetén z(darab)=x(i) : i=i+1 : j=j+1
        x(i)>y(j) esetén z(darab)=y(j) : j=j+1
    elágazás vége
ciklus vége
```

Összefésülés

Összefuttatás ütközővel (lásd az előző algoritmust), ha a két sorozatban biztosan nincs közös elem.

```
i=1
j=1
darab=0
x(n+1)=ütköző
y(m+1)=ütköző
ciklus amíg i<n+1 vagy j<m+1
    darab=darab+1
    ha x(i)<y(j) akkor
        z(darab)=x(i)
        i=i+1
    egyébként
        z(darab)=y(j)
        j=j+1
    elágazás vége
ciklus vége
```

Rendezések

Egyszerű cserés rendezés (*)

```
ciklus i=1-től n-1-ig
    ciklus j=i+1-től n-ig
        ha x(i)>x(j) akkor
            csere: x(i), x(j)
        elágazás vége
    ciklus vége
ciklus vége
```

Buborékos rendezés (*)

```
i=n
ciklus amíg i≥2
    utolsó=0
    ciklus j=1-től i-1-ig
        ha x(j)>x(j+1) akkor
            csere: x(j), x(j+1)
            utolsó=j
        elágazás vége
    ciklus vége
    i=utolsó
ciklus vége
```

Minimumkiválasztásos rendezés (*)

```
ciklus i=1-től n-1-ig
    min=i
    ciklus j=i+1-től n-ig
        ha x(min)>x(j) akkor
            min=j
        elágazás vége
    ciklus vége
    csere: x(i), x(min)
ciklus vége
```

Beillesztéses rendezés

```
ciklus i=2-től n-ig
    j=i-1
    temp=x(i)
    ciklus amíg j>0 és x(j)>temp
        x(j+1)=x(j) : j=j-1
    ciklus vége
    x(j+1)=temp
ciklus vége
```

Forrás: Szlávi Péter – Zsakó László: Módszeres programozás: Programozási tételek (Mikrológia 19, ELTE Informatikai Kar, 2004)