

Problémaosztályok, algoritmusok

3. Programozási tételek

Programozási tételek

Az algoritmusok osztályokba sorolása a bemenő és a kimenő adatok száma, illetve szerkezete, illetve a megoldandó feladat szerint:

- ▶ **Egy sorozathoz egy értéket rendelő** algoritmusok
- ▶ **Egy sorozathoz egy sorozatot rendelő** algoritmusok
- ▶ **Több sorozathoz egy sorozatot rendelő** algoritmusok
- ▶ **Egy sorozathoz több sorozatot rendelő** algoritmusok

Az egyes osztályokhoz feladat csoportok tartoznak. A feladatcsoportok megoldására mutatunk be algoritmusokat, melyeket programozási tételeknek is neveznek a szakirodalomban.



Egy sorozathoz egy értéket rendelő algoritmusok

▶ **Sorozatszámítás**

- ▶ Adott az $A[0], A[1], \dots, A[N-1]$ sorozat.
- ▶ Az adatok beolvasását itt és a továbbiakban nem részletezzük.
 - ▶ a./ Az összegzés algoritmus
 - ▶ b./ A szorzás algoritmus



Az összegzés algoritmus

Adott az $A[0], A[1], \dots, A[N-1]$ sorozat.

A sorozat elemei numerikus értékek. Határozzuk meg az elemek összegét, s -et.

Megoldás:

$$s = 0$$

Ciklus $cv = 0$ kezdőértéktől $cv = N-1$ végértékig 1 lépésközzel

$$s = s + a[cv]$$

Ciklus vége

Ki: s

Feladat: A fűtési szezonban naponta egyszer megmértük és feljegyeztük a nappali szoba hőmérsékletét. Mennyi volt az átlagos hőmérséklet ?



A szorzás algoritmus

Adott az $A[0], A[1], \dots, A[N-1]$ sorozat.

A sorozat elemei numerikus értékek. Határozzuk meg az elemek szorzatát, p -t.

Megoldás:

$$p = 1$$

Ciklus $cv = 0$ kezdőértéktől $cv = N-1$ végértékig 1 lépésközzel

$$p = p * a[cv]$$

Ciklus vége

Ki: p



Eldöntés

Adott az $A[0], A[1], \dots, A[N-1]$ sorozat és egy, a sorozat elemein értelmezett T tulajdonság. Eldöntendő, hogy van-e a sorozatban legalább egy T tulajdonságú elem.

Megoldás: A **van** nevű változó kezdeti értéke legyen nulla.

Sorban megvizsgáljuk a sorozat tagjait. Ha találunk olyan elemet, amely T tulajdonságú, akkor **van** értékét 1-re változtatjuk.

van = 0

ciklus $cv = 0$ kezdőértéktől $cv=N-1$ végértékig 1 lépésközzel

Ha $A[cv]$ T tulajdonság

van = 1

Elágazás vége

ciklus vége

Ha van

Ki: "Van a sorozatban T tulajdonságú elem"

Elágazás vége



Feladat: A fűtési szezonban naponta egyszer megmértük és feljegyeztük a nappali szoba hőmérsékletét. Volt-e olyan nap, amikor a hőmérséklet 18 fok alá csökkent ?

A megoldás pszeudo-kódja:

Program

Változók: **hofok [200]**

napok

cv

van

egész típusú tömb: a mért értékek

napok száma a fűtési szezonban

ciklusváltozó, egész típ.

0 a kezdeti érték, 1, ha volt 18-nál kisebb mért érték

Ciklus

Be: napok

//mérési értékek száma

amíg napok < 1 vagy napok > 200



Ciklus $cv = 0$ kezdőértéktől $cv < \text{napok}$ végértékig 1 lépésközzel

 Be: hofok [cv] // mért értékek bevitele

ciklus vége

van = 0 //az eldöntés algoritmus

Ciklus $cv = 0$ kezdőértéktől $cv < \text{napok}$ végértékig 1 lépésközzel

 Ha hofok [cv] < 18

 van = 1

 Elágazás vége

ciklus vége

for (cv=0; cv<napok; cv++)
 if (hofok[cv] < 18)
 van = 1;

Ha van

 Ki : "Volt 18 foknál alacsonyabb mért érték"

Elágazás vége

Program vége



Kiválasztás

- ▶ **Adott az $A[0], A[1], \dots, A[N-1]$ sorozat és egy, a sorozat elemein értelmezett T tulajdonság. Tudjuk, hogy van a sorozat elemei között legalább egy T tulajdonságú elem. A feladat az első ilyen T tulajdonságú elemnek, illetve a sorszámának a meghatározása. A feladat hasonló az eldöntéshez, de a kiválasztásnál más az output.**

- ▶ Megoldás: A *pozíció* változóban tároljuk az első T tulajdonságú elem indexét, a kezdeti értéke -1 .

(A van változó felesleges, mert *pozicio* = -1 értéke azt jelenti, hogy még nem találtunk T tulajdonságú elemet.)

Sorban megvizsgáljuk a sorozat tagjait. Ha találunk olyan elemet, amely T tulajdonságú, akkor a *pozicio* változó felveszi a T tulajdonságú elem index értékét.



Kiválasztás

A megoldás pszeudokódja:

pozicio = -1

cv = 0

ciklus amíg $cv < N$ ÉS $indext == -1$

 Ha $A[cv]$ T tulajdonságú

 pozicio = cv

 Elágazás vége

 cv = cv + 1

ciklus vége

ha $indext > -1$

 Ki: "Az elem és a sorszáma:", a [cv], cv



Feladat: A fűtési szezonban naponta egyszer megmértük és feljegyeztük a nappali szoba hőmérsékletét. Melyik volt az első nap, amikor a hőmérséklet 18 fok alá csökkent ?

Program

Változók: hofok [200] egész típ. tömb a mért értékek tárolására
napok napok száma a fűtési szezonban
cv ciklusváltozó, egész típ.
indext -1 a kezdeti érték,
 >=0 ha volt 18-nál kisebb mért érték

Ciklus //mérési értékek száma

 Be: napok

amíg napok < 1 vagy napok > 200

Ciklus cv = 0 kezdőértéktől cv = napok-1 végértékig 1 lépésközzel

 Be: hofok [cv] // mért értékek bevitele

ciklus vége



indext = -1 //a kiválasztás algoritmus

cv = 0

Ciklus amíg $cv < \text{napok} - 1$ és $\text{indext} == -1$

Ha $\text{hofok}[cv] < 18$

indext = cv

Elágazás vége

cv = cv + 1

ciklus vége

Ha $\text{indext} > -1$

Ki : " 18 foknál alacsonyabb mért érték volt :", indext,
a[indext]

Program vége



Lineáris keresés

- ▶ **Adott az $A[0], A[1], \dots, A[N]$ sorozat és egy, a sorozat elemein értelmezett T tulajdonság.**
- ▶ **EI kell dönteni, hogy van-e a sorozatban T tulajdonságú elem, és ha van, akkor meg kell adni az első ilyen elemet és az indexét. A feladat az eldöntés és a kiválasztás feladatát ötvözi egybe.**

Megoldás: Az elemeket meg kell vizsgálni sorban ugyanúgy, mint az eldöntés vagy a kiválasztás algoritmusánál. Az iteráció feltétele, hogy még nem találtunk T tulajdonságú elemet vagy még nem vizsgáltuk meg a sorozat összes elemét.



Lineáris keresés

A megoldás pszeudokódja:

indext = -1

cv = 0

ciklus amíg (indext == -1 ÉS cv < N)

 ha A[*cv*] T tulajdonságú

 indext = *cv*

 Elágazás vége

 cv = cv + 1

ciklus vége

ha indext > -1

 Ki: "Az elem és a sorszáma:", a[*cv*], *cv*

Elágazás vége



Megszámlálás

- ▶ **Adott az $A[0], A[1], \dots, A[N-1]$ sorozat és egy, a sorozat elemein értelmezett T tulajdonság.**
- ▶ **Meg kell számolni, hogy a sorozat hány eleme rendelkezik a T tulajdonsággal.**

Megoldás: A számlálásra az s változót használjuk, melynek kezdeti értéke 0. Az elemeket meg kell vizsgálni sorban és ha találunk T tulajdonságú elemet, az S értékét eggyel növeljük.

$s = 0$

Ciklus $cv = 0$ kezdőértéktől $cv = N-1$ végértékig 1 lépésközzel

Ha $A[cv]$ rendelkezik T tulajdonsággal

$s = s + 1$

Elágazás vége

ciklus vége

Ki : s



Maximum és minimum kiválasztás

- ▶ **Adott az N elemű $A[0], A[1], \dots, A[N-1]$ sorozat. Keressük meg a legnagyobb elemét. Írassuk ki a legnagyobb elem értékét és a sorszámát.**

Megoldás:

maxelem = $A[0]$

maxindex = 0

ciklus $cv = 1$ kezdőértéktől $cv = N-1$ végértékig 1 lépésközzel

Ha $A[cv] > \text{maxelem}$

maxelem = $A[cv]$

maxindex = cv

Elágazás vége

ciklus vége

Ki: maxelem, cv



Egy sorozathoz egy sorozatot rendelő algoritmusok

- ▶ **Kiválogatás**
- ▶ **Másolás**
- ▶ **Rendezés**



Kiválogatás

- ▶ Adott egy N elemű sorozat, $(A[0], A[1], \dots, A[N-1])$ és egy, a sorozat elemein értelmezett T tulajdonság. Ki kell gyűjteni a sorozat összes T tulajdonságú elemét. A kiválogatott elemeket egy B tömbben tároljuk.
- ▶ Az algoritmus abban különbözik a kiválasztás algoritmusától, hogy ott egyetlen T tulajdonságú elemet kell kiválasztani, itt pedig az összes T tulajdonságú elemet.

Megoldás:

$cv2 = 0$

ciklus $cv1 = 0$ kezdőértéktől $cv1 = N-1$ végértékig 1 lépésközzel

Ha $A[cv1]$ T tulajdonságú

$cv2 = cv2 + 1$

$B[cv2] = A[cv1]$

Elágazás vége

ciklus vége

Ki: b tömb elemei



Másolás

- ▶ **Adott az N elemű sorozat, $(A[0], A[1], \dots, A[N-1])$ amelynek alapján egy másik, ugyancsak N elemű sorozatot kell képezni úgy, hogy a sorozat elemein valamilyen átalakítást végzünk.**
- ▶ A sorozat tagjaihoz valamilyen szabály szerint rendelünk új értéket, jelöljük a transzformációt f -el. Az új sorozatot egy másik, B tömbben tároljuk.

ciklus $cv = 0$ kezdőértéktől $cv = N-1$ végértékig 1 lépésközzel

$$B[cv] = f(A[cv])$$

ciklus vége

- ▶ A transzformáció hiányozhat is akár, a feladattól függően.
-



Rendezés

- ▶ Adott az N elemű sorozat, amelynek elemeit növekvő, vagy csökkenő sorrendbe kell rendezni.
- ▶ A rendező algoritmusokkal a következő előadáson kiemelten foglalkozunk.



Több sorozathoz egy sorozatot rendelő algoritmusok

- ▶ **Egyesítés**
- ▶ **Metszetképzés**
- ▶ **Különbségképzés**
- ▶ **Rendezett sorozatok egyesítése**



Egyesítés

- ▶ **Adott két sorozat, $A[0], A[1], \dots, A[N-1]$ és $B[0], B[1], \dots, B[M-1]$, melyek nem tartalmaznak ismétlődő elemeket, vagyis a sorozatokat tekinthetjük halmazoknak. Képezzük a két halmaz unióját, melynek elemeit a C tömbben tároljuk.**

Megoldás:

- ▶ Az A tömb elemeit átmásoljuk a C tömbbe.
- ▶ Ezután a B tömb minden elemét megvizsgáljuk, hogy tagja-e az A tömbnek
- ▶ Ha nem eleme, akkor hozzáadjuk a C tömbhöz.



Metszetképzés

- ▶ **Adott két sorozat, $A[0], A[1], \dots, A[N-1]$ és $B[0], B[1], \dots, B[M-1]$, melyek nem tartalmaznak ismétlődő elemeket, vagyis a sorozatokat tekinthetjük halmazoknak. Képezzük a két halmaz metszetét, melynek elemeit a c tömbben tároljuk.**

Megoldás:

- ▶ Sorra vesszük az A tömb elemeit.
- ▶ Ha az adott elem a B tömbben is megtalálható, akkor bekerül a C tömbbe.



Különbségképzés

- ▶ **Adott két sorozat, $A[0], A[1], \dots, A[N-1]$ és $B[0], B[1], \dots, B[M-1]$, melyek nem tartalmaznak ismétlődő elemeket, vagyis a sorozatokat tekinthetjük halmazoknak. Képezzük a két halmaz különbségét, melynek elemeit a c tömbben tároljuk.**

Megoldás:

- ▶ Sorra vesszük az A tömb elemeit.
- ▶ Ha az adott elem a B tömbben NEM található, akkor bekerül a C tömbbe.



Rendezett sorozatok egyesítése

- ▶ **Adott két rendezett sorozat. Állítsunk elő belőlük szintén rendezett sorozatot úgy, hogy az eredeti sorozatok minden eleme szerepeljen benne.**
- ▶ Ha az eredeti sorozatok halmazok elemei (minden elemük különböző), és az egyesítéssel is halmazt kívánunk előállítani, akkor a két sorozat azonos elemeit csak egyszer vesszük figyelembe.
- ▶ A rendezett halmazok egyesítését összefuttatási feladatnak nevezik a számítástechnikában.



Rendezett sorozatok egyesítése

- ▶ **Megoldás:** Az **A** és **B** tömböt fésüljük össze **C**-be
 - ▶ IndA , IndB , ill. indC az **A**, **B** ill. a **C** sorozat indexe
 - ▶ Az aktuális elemek ($A[\text{IndA}]$ és $B[\text{IndB}]$) közül a kisebbet beletesszük a **C**-be
 - ▶ Növeljük a kiválasztott sorozat indexét és az indC -t
 - ▶ Ha az egyik sorozat végére érünk, a másik sorozat maradékát a **C**-be másoljuk.



Rendezett sorozatok egyesítése

IndA=0 ; IndB=0

Ciklus amíg IndA<N és IndB<M

 ha A[IndA]<B[IndB]

 C[indC]=A[IndA]

 IndA++

 különben

 C[IndC]=B[IndB]

 IndB++

 ha vége

 IndC++

Ciklus vége

Ciklus amíg IndA < N

 C[IndC]=A[IndA];

 IndC++;IndA++;

Ciklus vége

Ciklus amíg IndB < M

 C[IndC]=B[IndB];

 IndC++; IndA++;

Ciklus vége



Egy sorozatból egy sorozatot előállító algoritmusok

Szétválogatás

- ▶ **Adott az $a[1], a[2], \dots, a[n]$ sorozat és egy T tulajdonság, amelynek alapján a sorozatot két részsorozatra bontjuk.**

MEGOLDÁS:

- ▶ Azokat az elemeket, amelyek rendelkeznek a T tulajdonsággal, a b tömb első felébe másoljuk, növekvő index szerint,
- ▶ Amelyek pedig nem rendelkeznek a T tulajdonsággal, a b tömb második felébe tároljuk, csökkenő index szerint.

Az így a kapott sorozatok tárolására nem kell két tömböt lefoglalni, melyeknek nem tudjuk előre a hosszát.



Feladatok

- ▶ Implementáljátok a bemutatott algoritmusokat C nyelvben
- ▶ Írjatok programot amely összefésül három rendezett sorozatot
- ▶ Írjatok programot, amely helyben szétválogat egy sorozatot, az alapján hogy az adott elem nagyobb vagy kisebb mint a tömb első eleme
- ▶ Van egy tömb, amely hőmérsékleti adatokat tartalmaz. Határozzátok meg hogy melyik volt a legkisebb és legnagyobb hőmérséklet ill. hogy melyik napokon voltak ezek; az átlaghőmérsékletet, az első fagyos napot (amikor a hőmérséklet 0 fok alatt volt) és a fagyos napok számát.

