

DEVINTAS SKYRIUS

Rikiavimas

Tam, kad būtų lengviau rasti duomenį dideliame masyve, duomenys surikiuojami tam tikra tvarka: skaičiai didėjančiai arba mažėjančiai, žodžiai – pagal abėcėlę ir pan. Duomenis rikiuoti ir jų ieškoti tenka labai dažnai. Todėl ir algoritmų yra daug ir įvairių. Apie juos galima rasti beveik kiekviename programavimo vadovėlyje.

Rikiavimo ir paieškos metodai labai gerai aprašyti B. Burgio ir kt. knygoje „Kompiuterika moksleiviams ir studentams“ (Kaunas: Technologija, 1993, 1994), tryliktame (Paieška masyve) ir keturioliktame (Masyvų rikiavimas) skyriuose.

Be to, naudinga išsinaugrinėti 5, 6, 9 ir 52 uždavinius iš V. Dagienės ir J. Skūpienės knygos „Moksleivių informatikos olimpiadų uždaviniai: I–VII olimpiados“ (Vilnius: TEV, 1999).

Atkreipkite dėmesį, kad greitojo rikiavimo algoritmas bus nagrinėjamas kitame skyriuje, todėl jo naudoti šio skyriaus uždavinių sprendimuose negalima.

Uždaviniai

1. Masyvui rikiuoti nemažėjimo tvarka išrinkimo būdu parašytas algoritmas:

```
program rikiavimas;
const n = 20;
type intervalas = 1..n;
      masyvas = array [intervalas] of intervalas;
var A: masyvas;
      mažiausias, indeksas, ilgis, i, j: intervalas;
begin
  randomize;
  write('Įveskite masyvo ilgį nedidesnį, nei 20: ');
  readln(ilgis);
  writeln('Nesurikiuotas masyvas:');
  for i := 1 to ilgis do
    begin
      A[i] := 1 + random(20);
      write(A[i]:3)
    end;
  writeln;
  .
  .
  writeln('Surikiuotas masyvas');
  for i := 1 to ilgis do write(A[i]:3);
  readln
end.
```

Užbaikite programą *rikiavimas* (vietoj daugtaškio įrašykite reikiamus veiksmus) bei papildykite ją komentarais.

2. Duota 100 kortelių, išdėstytų eile. Ant kiekvienos kortelės užrašyta po vieną skaitmenį. Ar galima korteles perdėlioti taip, kad nė vienas skaitmuo nebūtų toje pačioje vietoje? Reikia rasti bent vieną perdėliotų kortelių variantą.

Pagrįskite uždavinio sprendimą ir užrašykite sprendimo idėją.

3. Masyvui rikiuoti nemažėjimo tvarka „burbulo“ būdu parašytas algoritmas:

```
program rikiavimas;
const n = 20;
type intervalas = 1..n;
```

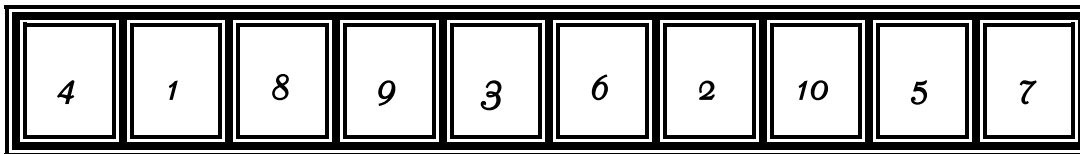
```

    masyvas = array [intervalas] of intervalas;
var A: masyvas;
    mažiausias, indeksas, ilgis, j: intervalas;
    keitimas: boolean;
    i: 0..n;
begin
    randomize;
    write('Įveskite masyvo ilgį nedidesnį, nei 20: ');
    readln(ilgis);
    writeln('Nesurikiuotas masyvas:');
    for i := 1 to ilgis do
        begin
            A[i] := 1 + random(20);
            write(A[i]:3)
        end;
    writeln;
    i := 0;
    repeat
        :
    until keitimas;
    writeln('Surikiuotas masyvas');
    for i := 1 to ilgis do write(A[i]:3);
    readln
end.

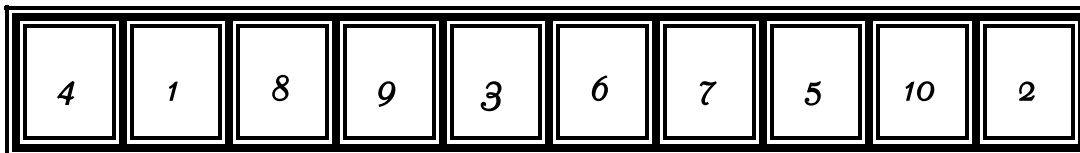
```

Užbaikite programą *rikiavimas* (vietoj daugtaškio įrašykite reikiamus veiksmus) bei papildykite ją komentarais.

4. Yra dešimt kortelių, sunumeruotų nuo 1 iki 10. Kortelės sudėliotos kaip parodyta paveiksle.



Kortelių tvarką galima keisti tik tokiu būdu: pasirenkama viena kortelė, pradedant ja visos dešinėje esančios kortelės sudėliojamos atvirkščia tvarka. Pavyzdžiui, jei pasirinksime kortelę „2“ ir atliksime nurodytus veiksmus, kortelės bus sudėliotos taip, kaip parodyta šiame paveikslėlyje:



Nurodykite, kurias korteles ir kokia tvarka reikia pasirinkti, kad pakeitus kortelių išdėstymą ankščiau aprašytu būdu, kortelės būtų sudėliotos jų numerių didėjimo tvarka.

Rezultatą turi sudaryti pasirenkamų kortelių numerių seka.

5. Masyvui rikiuoti nemažėjimo tvarka įterpimo būdu parašytas algoritmas:

```

program rikiavimas;
const n = 20;
type intervalas = 1..n;
    masyvas = array [intervalas] of intervalas;
var A: masyvas;
    skaičius, indeksas, ilgis, j: intervalas;
    keitimas: boolean;
    i: 0..n;
begin
    randomize;
    write('Įveskite masyvo ilgį nedidesnį, nei 20: ');
    readln(ilgis);
    writeln('Nesurikiuotas masyvas:');

```

```

for i := 1 to ilgis do
  begin
    A[i] := 1 + random(20);
    write(A[i]:3)
  end;
writeln;
:
:
writeln('Surikiuotas masyvas');
for i := 1 to ilgis do write(A[i]:3);
readln
end.

```

Užbaikite programą *rikiavimas* (vietoj daugtaškio įrašykite reikiamus veiksmus) bei papildykite ją komentarais.

6. Masyvą *PirmasMasyvas* = (7, 1, 2, 3, 4, 5, 6) procedūra, atliekanti „burbulo“ rikiavimo algoritmą, surikiuos nemažėjimo tvarka vienu žingsniu, o masyvą *AntrasMasyvas* = (2, 3, 4, 5, 6, 7, 1) – šešiais žingsniais.

Parašykite *procedūrą*, kurioje „burbulo“ metodas patobulintas: procedūroje turi būti atliekami masyvo apdorojimo krypčių pakeitimai, t. y. vieną kartą mažesni elementai stumiami į masyvo pradžią, o kitą kartą didesni elementai stumiami į masyvo pabaigą.

Šio skyriaus 1–6 uždavinių sprendimai turi būti pateikti JPM interneto svetainėje (<http://ims.mii.lt/jpm/>) iki 2008 m. spalio 19 d. 24 val.

Failo vardas turi būti ?????09.* (čia * – doc, txt, rtf arba odt). Vietoj klaustukų įrašykite pirmąsias penkias savo pavardės raides (be diakritinių ženklų). Jei kartais būtų pavardžių, trumpesnių, negu penkios raidės, trūkstanti simboliai keičiami pabraukimo brūkšniais „_“.

Elektroninio pašto adresas JPM antrosios dalies klausytojams: jpm.2kursas@gmail.com.

7. Nagrinėsime stačiakampius gretasienius. Laikysime, kad tikrai viena nagrinėjamų stačiakampių gretasienių viršūnė yra tame pačiame erdvės taške. Akivaizdu, kad į stačiakampį gretasienį, kurio briaunų ilgiai yra (6, 4, 9), telpa gretasienis (7, 4, 4), bet netelpa gretasienis (5, 6, 5).

Galime analogiškai apibendrinti uždavinį, imdami n ($n \geq 1$) matavimų erdvę. Laikoma, kad gretasienis $a = (a_1, a_2, \dots, a_n)$ telpa į gretasienį $b = (b_1, b_2, \dots, b_n)$, jeigu galima taip sutvarkyti gretasienio a briaunas ir gauti tokį pasuktą gretasienį $a' = (a'_1, a'_2, \dots, a'_n)$, kad būtų tenkinama sąlyga $a'_i \leq b_i$ ($1 \leq i \leq n$).

Tarkime, kad n -matėje erdvėje duota m stačiakampių gretasienių bei gretasienis g .

Parašykite: **a)** sprendimo idėją ir

b) programą, randančią gretasienių, telpančių į gretasienį g , skaičių.

Pradiniai duomenys įrašyti faile *duom.txt*. Pirmoje failo eilutėje įrašyti du natūralieji skaičiai: erdvės matas n ($1 \leq n \leq 20$) bei gretasienių skaičius m ($1 \leq m \leq 100$). Antroje eilutėje įrašyta n natūraliųjų skaičių – gretasienio g briaunų ilgiai. Į kiekvieną iš likusių m eilučių įrašyta po n natūraliųjų skaičių. Tai kitų duotųjų gretasienių briaunų ilgiai. Skaičiai eilutėse vienas nuo kito atskiriami vienu tarpu.

Rezultatą spausdinkite į failą *rez.txt*.

Pradinių duomenų ir rezultatų pavyzdys

<i>Pradiniai duomenys</i>	<i>Rezultatas</i>
3 6 7 4 4 1 8 2 5 5 5 3 2 1 7 4 5 1 1 7 6 3 2	3

8. Yra nuspaltinta n tiesės atkarpų. Žinomos kiekvienos atkarpos kairiojo ir dešiniojo galo koordinatės L_i ir R_i , čia $i = 1, \dots, n$. *Parašykite programą*, randančią nuspaltintos tiesės dalies ilgį.

Pradiniai duomenys įrašyti faile *duom.txt*. Pirmoje failo eilutėje įrašytas natūralusis skaičius n ($1 \leq n \leq 1000$). Kitose n eilučių įrašyti du sveikieji skaičiai, tarpusavy atskirti vienu tarpu, reiškiantys atkarpų kairiojo ir dešiniojo galo koordinatės l ir r ($-100000 \leq l < r \leq 100000$).

Rezultatas – nuspaltintos tiesės dalies ilgis – įrašomas į failą *rez.txt*.

Pradinių duomenų ir rezultatų pavyzdys

<i>Pradiniai duomenys</i>	<i>Rezultatai</i>
5 -1 1 7 8 3 5 -2 2 4 6	8
<i>Paaiškinimas</i>	
Atkarpa (-1; 1) yra atkarpos (-2; 2) viduje, todėl jų bendras ilgis lygus atkarpos (-2; 2) ilgiui: 4. Atkarpos (3; 5) ir (4; 6) susikerta, todėl jų bendras ilgis lygus atkarpos (3; 6) ilgiui: 3. Nesikertančių atkarpų (-2; 2), (3; 6) ir (7; 8) ilgis lygus jų ilgių sumai: $4 + 3 + 1 = 8$.	

Kartu su 7-o ir 8-o uždavinio sprendimu turi būti pateikta *programa, generuojanti **atsitiktinį sąlygą atitinkantį pradinių duomenų rinkinį (atsitiktinį testą), kuriame būtinai turi būti nors vienas ieškomas objektas.***

Vertinant šių uždavinių sprendimus programos bus tikrinamos (testuojamos) su sąlygose pateiktais pradiniais duomenimis. *Tik tos programos*, kurios su šiais duomenimis duos teisingus rezultatus, bus tikrinamos su specialiai parengtais kontroliniais testais ir vertinamos balais.

Programai surinkus *bent pusę balų* (5), bus vertinami 7-o uždavinio sprendimo idėjos aprašas, kuris turi būti parašytas kaip *komentaras sprendimo faile prieš programą*, bei 8-o uždavinio sprendimo programavimo stilius (*kultūra*).

Šio skyriaus 7 ir 8 uždavinių sprendimai turi būti pateikti JPM interneto svetainėje (<http://ims.mii.lt/jpm/>) iki 2008 m. lapkričio 2 d. 24 val.

Sprendimų failų vardai turi būti *?????097.pas* ir *?????098.pas*, generuojančių programų (*generatorių*) – *?????097gen.pas* ir *?????098gen.pas*. Vietoj klausukų įrašykite pirmąsias penkias savo pavardės raides (be diakritinių ženklų). Jei kartais būtų pavardžių, trumpesnių, negu penkios raidės, trūkstanti simboliai keičiami pabraukimo brūkšniais „_“.

Elektroninio pašto adresas JPM antrosios dalies klausytojams: jpm.2kursas@gmail.com.