

DEŠIMTAS SKYRIUS

Greitasis rikiavimas. Paieška.

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

Uždaviniai

1. Aprašykite žodžiais greitojo rikiavimo algoritmą. Kurioje algoritmo vietoje ir kaip realizuojamas principas „skaidyk ir valdyk“?

2. Masyvo elementui, kuris lygus duotajam skaičiui, ieškoti parašytas algoritmas:

```
program paieška;
const max = 100;
type intervalas = 1..max;
    masyvas = array [intervalas] of integer;
var A: masyvas;
    i, n: intervalas;
    skaičius: integer;
begin
    randomize;
    write('Įveskite masyvo ilgį nedidesnį, nei 100: '); readln(n);
    writeln('Sugeneruotas masyvas:');
    for i := 1 to n do
    begin
        A[i] := - (maxint + 1) + random(2 * maxint + 2);
        write(A[i]:7)
    end;
    writeln;
    write('Įveskite duotą skaičių: '); readln(skaičius);
    :
    :
    readln
end.
```

Atliekant programą, sugeneruotas, pavyzdžiui, toks masyvas: –32241 –32420 4755 8707 –11960. Jei ieškoma skaičiaus 5, programa turi atspausdinti: Skaičiaus 5 masyve nėra. Jei ieškoma skaičiaus 8707, programa turi atspausdinti: Skaičiui 8707 lygus 4 masyvo elementas. Užbaikite programą *paieška* (vietoj daugtaškio įrašykite reikiamus veiksmus) bei papildykite ją komentarais.

3. Egzistuoja toks žaidimas. Žaidimo vedėjas užmena dešimtainį skaičių iš intervalo nuo 0 iki $2^n - 1$ (n – neneigiamas sveikasis skaičius). Žaidėjas bando įminti skaičių užduodamas kuo mažiau klausimų, į kuriuos vedėjas gali atsakyti tik *Taip* arba *Ne*. Be abejo, egzistuoja sėkmės faktorius, bet kai n pakankamai didelis, žaidėjas ieško strategijos, kurios dėka greičiau įmintų skaičių.

Pasirodo, žaidėjui optimalu taikyti panašų į dvejetainę paiešką algoritmą. Kodėl? Aprašykite tokį algoritmą žodžiais. Kuo jis skiriasi nuo dvejetainės paieškos algoritmo ir kodėl?

Įdomus faktas: tam tikru būdu interpretuojant vedėjo atsakymus į pateikiamus pagal aptartą algoritmą klausimus, galima sukonstruoti užminto skaičiaus išraišką dvejetainėje skaičiavimo sistemoje.

Paašškinkite, kaip reikia interpretuoti atsakymus į klausimus, kodėl būtent taip ir kaip konstruojamas dvejetainis skaičius.

4. Žinoma, kad tarp 48 vizualiai vienodų monetų yra suklastota moneta – jos svoris kitoks. Bet nežinoma, ar jos svoris didesnis, ar mažesnis. Yra lygiapetės svirtinės svarstyklės. Ar galima nustatyti, kuri moneta suklastota, atlikus tik 4 svėrimus? Jei taip, aprašykite žodžiais svėrimų algoritmą.

5. Parašykite *procedūrą*, rikiuojančią nedidėjančia tvarka `integer` tipo elementų masyvą greitojo rikiavimo būdu.

6. Tarkime, turime pirminių skaičių masyvą. Parašykite *funkciją*, realizuojančią dvejetainės paieškos algoritmą tokiaame masyve. Funkcijos rezultatas – masyvo elemento, kuris lygus funkcijos argumentui, numeris arba 0.

Šio skyriaus 1–6 uždavinių sprendimai turi būti pateikti JPM interneto svetainėje (<http://ims.mii.lt/jpm/>) iki 2008 m. gruodžio 5 d. 24 val.

Failo vardas turi būti ?????10.* (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. Kelyje su skiriamąja juosta įrengta pėsčiųjų perėja su keturiais šviesoforais, po du kiekvienai važiuojamajai daliai. Žalia šviesa pėstiesiems atitinkamoje važiuojamojoje dalyje užsidega tą pačią sekundę, paspaudus šviesoforo mygtuką, ir dega lygiai trisdešimt sekundžių. Duomenys apie mygtukų paspaudimo momentus kaupiami Kelių eismo reguliavimo centre.

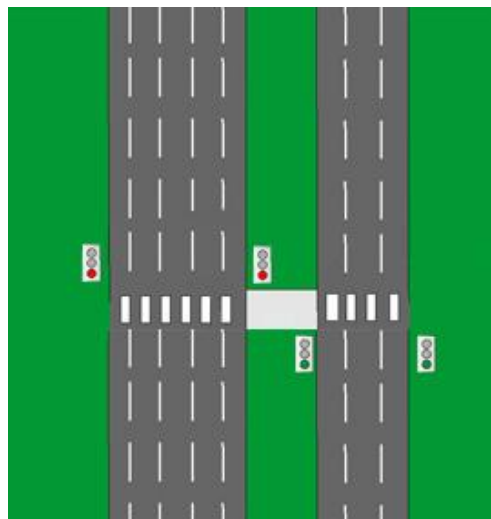
Parašykite: a) *sprendimo idėją* ir

b) *programą*, randančią, kiek sekundžių per vieną parą visuose šviesoforuose degė žalia šviesa.

Pradiniai duomenys įrašyti faile *duom.txt*. Pirmoje failo eilutėje įrašytas natūralusis skaičius n ($1 \leq n \leq 1000$). Kitose n eilučių įrašyti tokie duomenys, atskirti tarpu: raidė *D* arba *K* važiuojamajai daliai žymėti (*D* – dešinioji, *K* – kairioji); trys neneigiami sveikieji skaičiai *val*, *min* ir *sek*, rodantys paspaudimo laiką, išreikštą valandomis, minutėmis ir sekundėmis ($0 \leq val \leq 23$, $0 \leq min \leq 59$, $0 \leq sek \leq 59$).

Rezultatas – laikas sekundėmis – įrašomas į failą *rez.txt*.

Pradinių duomenų ir rezultatų pavyzdys



Pradiniai duomenys	Rezultatai	Paaiškinimas
8 K 21 1 59 D 7 8 0 K 13 55 13 D 13 54 59 D 21 2 29 K 7 8 0 D 7 7 50 K 13 55 30	44	07:08:00 užsidegė žalia šviesa abiejose važiuojamosiose dalyse ir degė 30 sekundžių. Dešinėsios dalies šviesoforai degė iki 13:55:29, o kairiosios užsidegė 13:55:13. Kartu degė 14 sekundžių. Kiti laikai nesikerta.

8. Duotas sąrašas, kuriame surašyti automobilių savininkai bei jų automobilių numeriai. Automobilio numerį sudaro trys lotyniškos raidės ir trys skaitmenys, sudarantys skaičių iš intervalo 1..999.

Sąrašą nusako toks duomenų tipas:

```

const SK = 1000;
type savin = record
    pavardė: string[20];
    num_raid: string[3]; { raidės ir }
    num_sk: 1..999       { skaitmenys, sudarantys automobilio numerį }
end;
sąrašas = array [1..SK] of savin;

```

Sąrašas sutvarkytas automobilių numerių didėjimo tvarka:

```

AAA001
AAA002
AAA005
AAA006
...
AAA999
AAB002
...

```

Parašykite programą, kuri pagal duotą automobilio numerį nustatytų jo savininko pavardę.

Pradiniai duomenys įrašyti faile *duom.txt*. Pirmoje failo eilutėje įrašytas numeris automobilio, kurio savininko pavardę reikia rasti. Antroje eilutėje įrašytas sąraše esančių automobilių skaičius n ($1 \leq n \leq 1000$). Į kiekvieną iš likusių n eilučių įrašyta po vieną automobilio savininko pavardę ir automobilio numerį.

Rezultatas – automobilio savininko pavardę – įrašomas į failą *rez.txt*.

Pradinių duomenų ir rezultatų pavyzdys

<i>Pradiniai duomenys</i>	<i>Rezultatas</i>
KVL578 6 Petraitis ABC321 Petkevičienė BJU500 Jonaitis DKG512 Antanaitis KVL578 Žemaitytė KVL644 Bartkus VVV255	Antanaitis

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 vertinama 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. gruodžio 18 d. 24 val.

Sprendimų failų vardai turi būti *?????107.pas* ir *?????108.pas*, generuojančių programų (*generatorių*) – *?????107gen.pas* ir *?????108gen.pas*. Vietoj klausukų įrašykite pirmąsias penkias savo pavardės raides (be diakritinių ženklų). Jei kartais būtų pavardžių, trumpesnių, negu penkios raidės, trūkstami simboliai keičiami pabraukimo brūkšniais „_“.

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