/\*duotas dvizenklis skaicius x. parasykite algoritma, kuris rastu skaiciaus x skaitmenis.\*/

#include <iostream>

using namespace std;

int main()

{

int

x, //pradinis skaicius

x1, //2-asis skaicius

x2; //3-asis skaicius

cout <<"įveskite dviženklį skaičių";

cin >> x;

x1 = x / 10;

x2 = x % 10;

cout << x1 <<" " << x2;

return 0;

}

/\*duotas dvizenklis skaicius x. parasykite algoritma, kuris rastu skaiciaus atbulai \*/

#include <iostream>

using namespace std;

int main()

{

int

x, //pradinis skaicius

x1, //2-asis skaicius

x2; //3-asis skaicius

cout <<"įveskite dviženklį skaičių";

cin >> x;

x1 = x / 10;

x2 = x % 10;

x = x2 \* 10 + x1;

cout << "Naujas skaicius yra" << x;

return 0;

}

/\*duotas dvizenklios skaicius t. parasykite algoritma kuris perrasytu si skaiciu taip, kad jo antrasis skaitmuo butu pirmuoju,trciasis antruoju, o pirmasis treciuoju pvz: 587 > 875\*/

#include <iostream>

using namespace std;

int main()

{

int t, t1, t2, t3;

cout << "įveskite triženklį skaičių";

cin >> t;

t1 = t /100;

t2 = (t /10) % 10;

t3 = t % 10;

t = t2 \* 100 + t3 \* 10 + t1;

cout << "naujas skaičius" << t;

return 0;

}

/\*Duotas keturženklis skaičius x. Parašykite algoritmus, kurie rastų skaičiaus x:

a) skaitmenis;

b) skaitmenų sumos kvadratą;

c) skaitmenų kvadratų sumą;

d) skaitmenų sumos kubą;

e) skaitmenų kubų sumą;

f) parašytų skaičių atbulai.\*/

#include <iostream>

using namespace std;

int main()

{

int x, x1, x2, x3, x4;

cout << "įveskite keturženklį skaičių";

cin >> x;

x1 = x /1000;

x2 = (x /100) % 10;

x3 = (x /10) % 10;

x4 = x % 10;

x = x1 + x2 + x3 + x4;

x = x \* x;

cout << "naujas skaičius" << x;

return 0;

}

/\*Duotas keturženklis skaičius x. Parašykite algoritmus, kurie rastų skaičiaus x:

a) skaitmenis;

b) skaitmenų sumos kvadratą;

c) skaitmenų kvadratų sumą;

d) skaitmenų sumos kubą;

e) skaitmenų kubų sumą;

f) parašytų skaičių atbulai.\*/

#include <iostream>

using namespace std;

int main()

{

int x, x1, x2, x3, x4;

cout << "įveskite keturženklį skaičių ";

cin >> x;

x1 = x /1000;

x2 = (x /100) % 10;

x3 = (x /10) % 10;

x4 = x % 10;

x = (x1 \* x1 \* x1) + (x2 \* x2 \* x2) + (x3 \* x3 \* x3) + (x4 \* x4 \* x4);

x = x1 + x2 + x3 + x4;

cout << "naujas skaičius" << " " << x;

return 0;

}  
  
/\*Žinomi kambario matmenys - ilgis ir plotis. Abu dydžiai yra integer. Reikia

apskaičiuoti, kokią pinigų sumą psuma reikės mokėti už plyteles, skirtas kambario

grindims iškloti, jei plytelių vieno kv m kaina yra m2kaina litų. Plytelių reikia

pirkti 5% daugiau galimiems nuostoliams padengti.\*/

#include <iostream>

using namespace std;

int main()

{ int ilgis, plotis, plotas;

double m2kaina, psuma;

cout << "Įveskite kambario ilgį: " ;

cin >> ilgis;

cout << "Įveskite kambario plotį: " ;

cin >> plotis;

plotas = ilgis \* plotis;

cout << "Kambario plotas yra: " << plotas<< endl;

cout << "kvadratinio metro kaina: ";

cin >> m2kaina;

psuma = 1.05 \* plotas \* m2kaina;

cout << " reikalinga pinigų suma: " << psuma;

return 0;

}

/\*1. Parašykite programą, kuri apskaičiuotų, kiek popieriaus lapų k reikės norint nukopijuoti knygą visos klasės mokiniams. Žinoma, kad klasėje yra n mokinių ir knygą sudaro m lapų.

Pasitikrinkite: Įvedę n = 20 ir m = 30, turėtume gauti k = 600.

.\*/

#include <iostream>

using namespace std;

int main()

{ int k, n, m;

cout << "Įveskite mokinių skaičių: " ;

cin >> n;

cout << "Įveskite lapų skaičių: " ;

cin >> m;

k = n \* m;

cout << "Reikiamų lapų skaičius: " << k;

return 0;

}

/\*2. Laikrodis rodo x valandų ir y minučių.

Parašykite programą, kuri apskaičiuotų, kiek minučių m ir kiek sekundžių s prabėgo nuo vidurnakčio.

Pasitikrinkite: Įvedę x = 3 ir y = 5, turėtume gauti: m = 185, s = 11100.\*/

#include <iostream>

using namespace std;

int main()

{ int x, y, m, s;

cout << "Įveskite valandų skaičių: " ;

cin >> x;

cout << "Įveskite minučių skaičių: " ;

cin >> y;

m = (x \* 60) + y;

s = (x \* 3600) + (y \* 60);

cout << "Minučių skaičius: " << m << endl;

cout << "Sekundžių skaičius: " << s << endl;

return 0;

}  
  
  
  
**Užduotis**. Rasti kvadratinės lygties ax2 + bx + c = 0 sprendinius. Čia a, b, c – skaičiai, nelygūs nuliui.

**Algoritmas**. Kvadratinės lygtys sprendžiamos taip:

* Skaičiuojamas diskriminantas: d = b2 - 4ac.
* Tikrinama, ar lygtis turi sprendinių ir jei taip, tai jie skaičiuojami:
  + jei d < 0, kvadratinė lygtis neturi realių sprendinių;
  + jei d = 0, tuomet kvadratinė lygtis turi vieną sprendinį ;
  + jei d > 0, tuomet kvadratinė lygtis turi du sprendinius:

/\*Užduotis. Rasti kvadratinės lygties ax2 + bx + c = 0 sprendinius. Čia a, b, c – skaičiai, nelygūs nuliui.

Algoritmas. Kvadratinės lygtys sprendžiamos taip:

Skaičiuojamas diskriminantas: d = b2 - 4ac.

Tikrinama, ar lygtis turi sprendinių ir jei taip, tai jie skaičiuojami:

jei d < 0, kvadratinė lygtis neturi realių sprendinių;

jei d = 0, tuomet kvadratinė lygtis turi vieną sprendinį ;

jei d > 0, tuomet kvadratinė lygtis turi du sprendinius: ir .\*/

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{ int a, b, c; // lygties koeficientai.

double d; // diskriminantas.

double x1, x2; //saknys lygties.

cout << "Įveskite lygties koeficientus: " ;

cin >> a >> b >> c;

//skaiciavimai ir rezultatu isvedimai

d = b \* b - 4 a \* c;

if (d < 0) cout << "Lygtis sprendinių neturi. ";

else if (d == 0)

{

x1 = -b / 2 \* a;

cout << " lygtis turi vieną sprendinį " << x1;

}

else

{

x1 = (-b - sqrt(d)) / (2\*a);

x2 = (-b + sqrt(d)) / (2\*a);

cout << " Lygtis turi 2 sprendinius"

<< setw(6) << fixed << setprecision(2) << x1 << " "

<< setw(6) << fixed << setprecision(2) << x2 ;

}

return 0;

Užduotis.

Pakeiskite programą, kad ji spręstų kvadratines lygtis su realiojo tipo koeficientais.   
  
/\*Pakeiskite programą, kad ji spręstų kvadratines lygtis su realiojo tipo koeficientais.\*/

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{ double a, b, c; // lygties koeficientai.

double d; // diskriminantas.

double x1, x2; //saknys lygties.

cout << "Įveskite lygties koeficientus: " ;

cin >> a >> b >> c;

//skaiciavimai ir rezultatu isvedimai

d = b \* b - 4 a \* c;

if (d < 0) cout << "Lygtis sprendinių neturi. ";

else if (d == 0)

{

x1 = -b / 2 \* a;

cout << " lygtis turi vieną sprendinį " << x1;

}

else

{

x1 = (-b - sqrt(d)) / (2\*a);

x2 = (-b + sqrt(d)) / (2\*a);

cout << " Lygtis turi 2 sprendinius"

<< setw(6) << fixed << setprecision(2) << x1 << " "

<< setw(6) << fixed << setprecision(2) << x2 ;

}

return 0;  
  
  
Bendras pažymys

/\*3. Šiandien Kęstutis švenčia gimtadienį. Jam sukanka a metų.

Parašykite programą, kuri apskaičiuotų, kiek mėnesių men, dienų d ir valandų v Kęstutis gyvena šiame pasaulyje.

Tarkime, kad metai turi 365 dienas.

Pasitikrinkite: Įvedę a = 16, turėtume gauti: men = 192, d = 5840, v = 140160.\*/

/\*Šviesoforas veikia pagal tokį algoritmą:

kiekvienos valandos pirmąsias tris minutes dega žalia šviesa, po to dvi minutes raudona, po to vėl tris minutes žalia ir t.t.

Žinoma, kiek minučių t praėjo nuo valandos pradžios.

Parašykite programą, kuri nustatytų, kokia šviesa dega.\*/

#include <iostream>

using namespace std;

int main()

{ int t;

cout << "įveskite minuciu skaaiciu ";

cin >> t;

if (t % 5 == 4)

cout << "dega raudona";

else if (t % 5 == 0)

cout << "dega raudona, tuoj bus zalia";

else if (t % 5 == 3)

cout << "dega zalia, tuoj bus raudona";

else cout << "dega žalia";

return 0;

}

/\*3. Šiandien Kęstutis švenčia gimtadienį. Jam sukanka a metų.

Parašykite programą, kuri apskaičiuotų, kiek mėnesių men, dienų d ir valandų v Kęstutis gyvena šiame pasaulyje.

Tarkime, kad metai turi 365 dienas.

Pasitikrinkite: Įvedę a = 16, turėtume gauti: men = 192, d = 5840, v = 140160.

\*/

#include <iostream>

using namespace std;

int main()

{ int a, men, d, y;

cout << "įveskite metu skaaiciu ";

cin >> a;

men = a \* 12;

d = a \* 365;

y = a \* (365 \* 24);

cout << "men = " << men << endl;

cout << "d = " << d << endl;

cout << "y = " << y << endl;

return 0;

}

/\*4. Parašykite programą, kuri apskaičiuotų, kiek knygų k vidutiniškai per metus perskaito vienas mokyklos bibliotekos lankytojas.

Žinomas vidutiniškai per vieną mėnesį perkaitytų knygų skaičius v ir vidutiniškai per metus apsilankiusiųjų bibliotekoje skaičius n.

Pasitikrinkite: Įvedę v = 120 , n = 800, turėtume gauti: k = 2;

\*/

#include <iostream>

using namespace std;

int main()

{ int k, v, n;

cout << "įveskite vidutinį lankytojų skaaiciu per metus ";

cin >> n;

cout << "Įveskite vidutiniskai perskaitytu knygu per menesi kieki ";

cin >> v;

k = (v \* 12) / n;

cout << "vidutinis knygu skaicius per metus: " << k;

return 0;

}

11.06:**Užduotis**. Elektros laidininką sudaro n nuosekliai sujungtų laidininkų, kurių varžos yra r1, r2, ..., rn omų. Reikia apskaičiuoti grandinės varžą r.

**Uždavinio sprendimo algoritmas:**

Tarkime, kad elektros grandinę sudaro n = 4 nuosekliai sujungti laidininkai, kurių varžos yra r1 = 2, r2 = 4, r3 = 1, r4 = 4. Grandinės varža skaičiuojama sumuojant visų laidininkų varžas. Žinodami laidininkų skaičių n ir kiekvieno laidininko varžą rlaid, grandinės varžą r galite rasti pagal algoritmą, kurio veiksmai kartojami n kartų. Prieš pradedant vykdyti veiksmus, būtina žinoti n reikšmę, o kintamojo r pradinė reikšmė yra lygi nuliui. Atliekami tokie pasikartojantys skaičiavimai:

· įvedama laidininko varža rlaid;

· skaičiuojama grandinės varža r = r + rlaid;.

Lentelėje pavaizduota, kaip atliekami veiksmai:

| **Ciklas vykdomas i-tąjį kartą** | **Įvedama i-tojo laidininko varža** | **Skaičiuojama grandinės varža** r = r + rlaid; |
| --- | --- | --- |
| 1 | 2 | r = 0 + 2 = 2 |
| 2 | 4 | r = 2 + 4 = 6 |
| 3 | 1 | r = 6 + 1 = 7 |
| 4 | 4 | r = 7 + 4 = 11 |

Skaičiavimų rezultatas: r = 11.

/\* Elektros laidininką sudaro n nuosekliai sujungtų laidininkų, kurių varžos yra r1, r2, ..., rn omų.

Reikia apskaičiuoti grandinės varžą r.\*/

#include <iostream>

using namespace std;

int main()

{

int n; //laidininkų skaičius.

double rlaid; // laidininko varža

double r; //grandinės varža

cout << "Kiek yra laidininkų?" << endl;

cin >> n;

r = 0;

for (int i = 1; i <= n; i++)

{

cout << "įveskite laidinko varžą: "<< i << "-ojo laidininko varža: ";

,

cin >> rlaid;

r = r + rlaid; //r += rlaid;

}

cout << "Varža" << setw(6) << fixed << setprecision (2) << r << endl;

return 0;

}

1. Klasėje mokosi n mokinių. Kiekvieno mokinio ūgis yra u1, u2, ..., un centimetrų. Parašykite programą, skaičiuojančią vidutinį klasės mokinių ūgį uvid.

*Pasitikrinkite, kai n = 5, o u1 = 179, u2 = 180, u3 = 178, u4 = 179, u5 = 175, turi būti spausdinama: Vidutinis klasės mokinio ūgis uvid = 178.20 cm.*

2. Šachmatų išradėjas iš valdovo paprašė tokio atlygio: ant pirmo šachmatų lentos langelio padėkite vieną grūdą, ant antro – du, ant trečio – keturis ir t. t. vis dvigubinkite, kol pasibaigs langeliai. Valdovas tik nusijuokė ir paliepė atseikėti grūdų. Kiek grūdų gaus šachmatų išradėjas? Šachmatų lentoje yra 64 langeliai. Parašykite programą šiam uždaviniui spręsti. Uždaviniui spręsti panaudokite real duomenų tipą rezultato reikšmei saugoti.

/\* 1.Klasėje mokosi n mokinių. Kiekvieno mokinio ūgis yra u1, u2, ..., un centimetrų.

Parašykite programą, skaičiuojančią vidutinį klasės mokinių ūgį uvid.

Pasitikrinkite, kai n = 5, o u1 = 179, u2 = 180, u3 = 178, u4 = 179, u5 = 175, turi būti spausdinama:

Vidutinis klasės mokinio ūgis uvid = 178.20 cm.\*/

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

int n;

double uvid;

double u;

double h;

cout << "Kiek yra mokinių? " << endl;

cin >> n;

u = 0;

for (int i = 1; i <= n; i++)

{

cout << "įveskite mokinų ugį: " ;

cin >> h;

u = (h+u)

r = r + rlaid;

}

uvid = u/5;

cout << "vidutinis mokinių ūgis: " << setw(6) << fixed << setprecision (2)<< r << endl;

return 0;

}

/\*2 Šachmatų išradėjas iš valdovo paprašė tokio atlygio: ant pirmo šachmatų lentos

langelio padėti vieną grūdą, ant II-ojo - du, ant III-iojo - keturis ir t.t., vis

dvigubinant, kol pasibaigs langeliai. Valdovas tik nusijuokė ir paliepė

atseikėti grūdų. Kiek grūdų gaus šachmatų išradėjas? Šachmatų lentoje yra 64 langeliai.

Parašykite programą šiam uždaviniui spręsti. Rezultatui naudokite real tipą.

Ats: Šachmatų išradėjas gaus 18446744073709550000 grūdų.\*/

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

double gr;

double viso;

gr = 1;

viso = 1;

for (int i = 2; i <= 64 ; i++)

{

gr = gr\*2;

viso = viso + gr;

}

cout << "Viso grūdų zmogus gaus: " << setw(50) << fixed << setprecision (0)<< viso << endl;

return 0;

}

/\*3 Architektas suprojektavo salę, kurioje bus n eilių.

Pirmoje eilėje stovės k kėdžių, o kiekvienoje kitoje eilėje -

2 kėdėmis daugiau, negu prieš tai buvusioje. Parašykite programą,

kuri apskaičiuotų, kiek iš viso kėdžių s reikia užsakyti, kad architekto sumanymas būtų įgyvendintas.

pasitikrinkite. Kai n=3, o k=8, turi būti spausdinama: s=30 kėdžių\*/

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

int n;

int k;

int s;

cout << "iveskite eiliu skaiciu: ";

cin >> n;

cout << "iveskite kedziu skaiciu: ";

cin >> k;

for (int i = 1; i <= n ; i++)

{

k= ;

s= n \* (k+2);

cout << i <<" "<<s<<endl;

}

cout << "Viso kedziu reikes: " << setw(10) << fixed << setprecision (0)<< s << endl;

return 0;

}

/\*2.6 Siena. Iš plytų galima pastatyti vienos plytos storio taisyklingą sieną, kurios

viršūnėje yra viena plyta, o šonuose - pusės plytos ilgio laipteliai.

Reikia parašyti programą, kuri apskaičiuotų, kelių plytų a bus sienos pagrindas

ir kelių plytų aukščio k bus siena, jei žinomas sienos statybai skirtų plytų

skaičius p.\*/

#include <iostream>

using namespace std;

int main()

{

int p, s, a, k;

cout << "Kiek plytu skirta sienos staytbai?: " << endl;

cin >> p;

a = 0; k = 0; s = 0;

cout <<"------------------------------------------------------"<<endl;

cout << "pagrindo plotis aukstis panaudotu plytu skaicius"<<endl;

cout <<"------------------------------------------------------"<<endl;return 0;

while ((a + 1) <= (p - s))

{

a ++;

k ++;

s += a;

cout << " " << a << " "<< k<< " "<< s << endl;

}

cout <<"------------------------------------------------------"<<endl;

cout << "sienos ilgis: "<<a<<endl;

cout << "sienos aukstis: "<<k<<endl;

cout << "panaudotu plytu skaicius: "<<s<<endl;

return 0;

}

{2.6.1Martynas labai mėgsta saldainius. Mamos slėptuvėje berniukas rado m saldainių.

Pirmą dieną jis suvalgė 1 saldainį, antrąją - 2, trečiają - 3. Kiekvieną kitą

dieną jis suvalgydavo vienu saldainiu daugiau negu prieš tai buvusią dieną.

Per kelias dienas d Martynas suvalgys visus saldainius? Paskutinei dienai gali

likti mažiau saldainių. Kai m=11, turėtume gauti d=5.}

{2.6.2 Bankas už indėlius moka p procentų palūkanų per metus. Metų gale palūkanos

pridedamos prie indėlio. Jei indėlininkas pinigų nė kiek neišima, palūkanos

skaičiuojamos nuo vis didesnės sumos. Parašykite programą, kuri apskaičiuotų,

per kiek metų t pradinis indėlis ind pasieks sumą s.}

{2.6.3 Turime kompiuterį, kuris nemoka apskaičiuoti natūraliųjų skaičių dalmens sveikosios dalies ir liekanos (nėra div ir mod dalybos operacijų).

Parašykite programą, kuri apskaičiuotų skaičių n ir m dalmens sveikąją dalį dalmuo ir liekaną liekana.

Kai n=14, m=3, turėtume gauti dalmuo=4, liekana=2}

{2.6.4 paskalio kalboje nėra funkcijos, skirtos realiajam skaičiui x kelti sveikuoju laipsniu k.

Parašykite programą šiam uždaviniu spręsti. Rezultatą y pateikite dviejų ženklų po

kablelio tikslumu. Kai x=5, k=2, turėtumėte gauti y=25,00}

{2.6.5 Pristigo žmogui pinigų ir nuėjo pasiskolinti jų iš kaimyno. Tas sutiko

paskolinti, bet paprašė grąžinti juos kitą mėnesį tokia tvarka: pirmą mėnesio

dieną - 1 litą, antrąją - 2 Lt, trečiąją - 4 Lt, t.y. kiekvieną dieną du kartus

daugiau negu prieš tai buvusią. Tai, kas bus atiduota daugiau, ir bus kaimyno

palūkanos. Parašykite programą, kuri apskaičiuotų, kiek palūkanų litais gaus

kaimynas už paskolintus n litų}

{2.6.6 Iš n kvadratinių plytelių reikia sudėlioti vienos plytelės storio kvadratus:

pirmiausia sudėti didžiausią galimą kvadratą, iš likusių plytelių - vėl didžiausią

ir t.t. Parašykite programą, kuri išskaidytų nurodytą plytelių skaičių į dalis,

reikalingas kiekvieno kvadrato statybai.}

{2.6.1Martynas labai mėgsta saldainius. Mamos slėptuvėje berniukas rado m saldainių.

Pirmą dieną jis suvalgė 1 saldainį, antrąją - 2, trečiają - 3. Kiekvieną kitą

dieną jis suvalgydavo vienu saldainiu daugiau negu prieš tai buvusią dieną.

Per kelias dienas d Martynas suvalgys visus saldainius? Paskutinei dienai gali

likti mažiau saldainių. Kai m=11, turėtume gauti d=5.}

{2.6.2 Bankas už indėlius moka p procentų palūkanų per metus. Metų gale palūkanos

pridedamos prie indėlio. Jei indėlininkas pinigų nė kiek neišima, palūkanos

skaičiuojamos nuo vis didesnės sumos. Parašykite programą, kuri apskaičiuotų,

per kiek metų t pradinis indėlis ind pasieks sumą s.}

{2.6.5 Pristigo žmogui pinigų ir nuėjo pasiskolinti jų iš kaimyno. Tas sutiko

paskolinti, bet paprašė grąžinti juos kitą mėnesį tokia tvarka: pirmą mėnesio

dieną - 1 litą, antrąją - 2 Lt, trečiąją - 4 Lt, t.y. kiekvieną dieną du kartus

daugiau negu prieš tai buvusią. Tai, kas bus atiduota daugiau, ir bus kaimyno

palūkanos. Parašykite programą, kuri apskaičiuotų, kiek palūkanų litais gaus

kaimynas už paskolintus n litų}

{2.6.6 Iš n kvadratinių plytelių reikia sudėlioti vienos plytelės storio kvadratus:

pirmiausia sudėti didžiausią galimą kvadratą, iš likusių plytelių - vėl didžiausią

ir t.t. Parašykite programą, kuri išskaidytų nurodytą plytelių skaičių į dalis,

reikalingas kiekvieno kvadrato statybai.}

/\*{2.6.1Martynas labai mėgsta saldainius. Mamos slėptuvėje berniukas rado m saldainių.

Pirmą dieną jis suvalgė 1 saldainį, antrąją - 2, trečiają - 3. Kiekvieną kitą

dieną jis suvalgydavo vienu saldainiu daugiau negu prieš tai buvusią dieną.

Per kelias dienas d Martynas suvalgys visus saldainius? Paskutinei dienai gali

likti mažiau saldainių. Kai m=11, turėtume gauti d=5.}\*/

#include <iostream>

using namespace std;

int main()

{

int m, d;

cout << "Įveskite saldainiu skaičių: " << endl;

cin >> m;

d = 0;

while (m>0)

{

d ++;

m = m - d;

}

cout << "Suvalgys per " << d << "dienų" << endl;

return 0;

}

/\*{2.6.2 Bankas už indėlius moka p procentų palūkanų per metus. Metų gale palūkanos

pridedamos prie indėlio. Jei indėlininkas pinigų nė kiek neišima, palūkanos

skaičiuojamos nuo vis didesnės sumos. Parašykite programą, kuri apskaičiuotų,

per kiek metų t pradinis indėlis ind pasieks sumą s.}\*/

#include <iostream>

using namespace std;

int main()

{

int p, t, s , ind;

cout << "Įveskite padetu pinigu skaičių: " << endl;

cin >> ind;

cout << " iveskite procentu skaiciu: "<< endl;

cin >> p;

while ( ind != s)

{

s = (ind \* p) \* t;

t = s \* p;

cout << "Suma: " << s;

cout << "laikas: " << t;

}

return 0;

}

### Funkcijos apibrėžimo srities tyrimas

Atlikdami šį darbą išmoksite tinkamai užrašyti sąlygos tikrinimo sakinį.

| **Nuorodos į Paskalio kalbos žinyną** | **Nuorodos į algoritmų žinyną** |
| --- | --- |
| 3.1. Kintamasis, kintamojo reikšmė  3.2. Priskyrimo sakinys  3.3. Duomenų įvedimas klaviatūra  3.4. Rezultatų (duomenų) išvedimas ekrane  3.5. Ciklo sakinys while  3.7. Sąlygos sakinys if  3.8. Funkcijų sąrašas | 4.1. Tiesinis algoritmas  4.2. Ciklinis algoritmas  4.3. Šakotas skaičiavimas |

**Užduotis**. Apskaičiuokite funkcijos reikšmę, kai mp m mg ir kinta žingsniu mz. Čia mp, mg, mz – realieji skaičiai.

**Komentaras**. Atkreipkite dėmesį, kad trupmenos vardiklis negali būti lygus nuliui, o pošaknis neigiamas.

**Pasiruošimas**. Sukurkite katalogą Darbas6 programos failams saugoti, atverkite FPS terpę ir sukurkite programos failą Darbas6.pas.

***Pirmas žingsnis****.* Pradinių duomenų įvedimas klaviatūra.

Ø Pradiniai duomenys:

mp – funkcijos argumento m pradinė reikšmė;

mg – funkcijos argumento m galinė (paskutinė) reikšmė;

mz – funkcijos argumento m kitimo žingsnis.

/\*Apskaičiuokite funkcijos reikšmę, kai mp m mg ir kinta žingsniu mz. Čia mp, mg, mz – realieji skaičiai.\*/

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

double mp, mg, mz, m, y;

cout << "Įveskite pradinę reikšmę" << endl;

cin >> mp;

cout << "Įveskite galinę reikšmę" << endl;

cin >> mg;

cout << "Įveskite žingsnio reikšmę" << endl;

cin >> mz; m = mp;

while (m <= mg)

{

if (m \* m - 100 > 0){

y = (m+3) / (sqrt(m\*m-100));

cout << setw (7) << fixed << setprecision(2) << m

<< setw (12) << fixed << setprecision(5) << y << endl;

}

else cout<<setw (7)<<fixed<<setprecision(2) << m

<<setw(12) <<fixed <<setprecision(0) << "\*\*\*\*\*\*\*"<<endl;

m += mz;

}

return 0;

/\*{2.6.5 Pristigo žmogui pinigų ir nuėjo pasiskolinti jų iš kaimyno. Tas sutiko

paskolinti, bet paprašė grąžinti juos kitą mėnesį tokia tvarka: pirmą mėnesio

dieną - 1 litą, antrąją - 2 Lt, trečiąją - 4 Lt, t.y. kiekvieną dieną du kartus

daugiau negu prieš tai buvusią. Tai, kas bus atiduota daugiau, ir bus kaimyno

palūkanos. Parašykite programą, kuri apskaičiuotų, kiek palūkanų litais gaus

kaimynas už paskolintus n litų}

\*/

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

int n, s, p, g;

cout << "Įveskite pinigu skaiciu" << endl;

cin >> n;

{

if (m \* m - 100 > 0){

y = (m+3) / (sqrt(m\*m-100));

cout << setw (7) << fixed << setprecision(2) << m

<< setw (12) << fixed << setprecision(5) << y << endl;

}

else cout<<setw (7)<<fixed<<setprecision(2) << m

<<setw(12) <<fixed <<setprecision(0) << "\*\*\*\*\*\*\*"<<endl;

m += mz;

}

return 0;

}

/\*2.7.1 Klasėje yra n mokinių. Jų it savarankiško darbo pažymiai yra p1, p2, .., pn.

Parašykite programą, kuri apskaičiuotų, kelių mokinių k darbai buvo

įvertinti 9 ir 10.\*/

#include <iostream>

using namespace std;

int main()

{

int n, m, k, p;

cout << "Įveskite mokiniu skaiciu: " << endl;

cin >> n;

m = 0;

k = 0;

while (m < n){

cout << "įveskite mokiniu pazymiu skaiciu:"<< endl;

cin >> p;

if (p >= 9)

k ++;

}

cout << " mokiniu kurie gavo 9 arba 10 skaicius: "<<k;

return 0;

}

/\*2.7.2 Parduotuvėje žmogus pirko n rūšių prekių. Jų kainos yra k1,k2,..,kn litų.

Parašykite programą, kuri apskaičiuotų, kiek yra prekių k, kurių kaina ne didesnė kaip m litų,

ir kokią pinigų sumą s reikės mokėti už šias prekes.

Pasitikrinkite, kai n=5,m = 14, k1=12, k2=6, k3=19, k4=16, k5=2, turi būti spausdinama:

„Prekių, kurių kaina ne didesnė kaip 14 litų, yra 3. Už jas reikės mokėti 20 Lt.“

\*/

#include <iostream>

using namespace std;

int main()

{

int n, m, s, k, z;

cout << "Įveskite prekiu skaiciu: " << endl;

cin >> n;

z = 0;

k = 0;

cout << "įveskite didžiausia galima kaina: "<<endl;

cin >> m;

while (z <= n)

{

z ++;

cout << " įveskite kainas: "<< endl;

cin >> k;

if (p >= 9)

k ++;

}

cout << " mokiniu kurie gavo 9 arba 10 skaicius: "<<k;

return 0;

}

/\*2.8 Trys lazdos

Atlikdami šį darbą susipažinsite su loginiais kintamaisiais ir loginiais reiškiniais, loginėmis operacijomis IR, ARBA.

Išmoksite loginius reiškinius panaudoti ciklo ir sąlygos sakiniuose.

Užduotis. Duota tam tikro ilgio lazdų n rinkinių.

Kiekvieną rinkinį sudaro 3 lazdos.

Lazdų ilgiai a, b ir c matuojami decimetrais (sveikieji skaičiai).

Nustatyti, ar galima iš šių lazdų sudėti trikampį. Jeigu galima, tai kokį: statųjį, lygiakraštį, lygiašonį ar įvairiakraštį.

Jeigu negalima, tai užrašyti atitinkamą tekstą.

Algoritmas. Sprendžiant šią užduotį galime įsivaizduoti, kad trys lazdos geometrijoje atitinka tris atitinkamo ilgio atkarpas a, b ir c. Prisiminkite, kad ne visuomet iš trijų atkarpų galima sudėti trikampį. Būtent, negalima sudėti trikampio, jeigu yra bent viena pora atkarpų, kurios ilgių suma yra mažesnė arba lygi už likusiąją trečiąją atkarpą. Arba galima pasakyti ir taip, kad iš trijų atkarpų trikampį galima sudėti tuomet, kai visų galimų (arba bet kurių) atkarpų porų ilgių suma yra didesnė už trečiąją atkarpą, t. y.

(a + b > c) ir (a + c > b) ir (c + b > a).\*/

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

int a, b, c;

cout << "įveskite kiek lazdu ilgius: " << endl;

cin >> a >> b >> c;

cout << "Lazdos: " << setw(2) << fixed << a << " "

<< setw(2) << fixed << b << " "

<< setw(2) << fixed << c << endl;

if ((a + b > c)&& (a + c > b) && (b + c > a)) //ar trikampis?

if ((a == b) && (b == c)) // ar lygiakrastis?

cout << "galima sudaryti lygiakrasti trikampį " << endl;

else if ((a == b) || (b == c) || (a == c)) //ar lygiasonis

cout << "galima sudaryti lygiašonį trikampį " << endl;

else cout << "galima sudaryti įvairiakraštį trikampį" <<endl;

else cout << "trikampio sudaryti negalima" << endl;

return 0;

}

/\*2.8 Trys lazdos

Atlikdami šį darbą susipažinsite su loginiais kintamaisiais ir loginiais reiškiniais, loginėmis operacijomis IR, ARBA.

Išmoksite loginius reiškinius panaudoti ciklo ir sąlygos sakiniuose.

Užduotis. Duota tam tikro ilgio lazdų n rinkinių.

Kiekvieną rinkinį sudaro 3 lazdos.

Lazdų ilgiai a, b ir c matuojami decimetrais (sveikieji skaičiai).

Nustatyti, ar galima iš šių lazdų sudėti trikampį. Jeigu galima, tai kokį: statųjį, lygiakraštį, lygiašonį ar įvairiakraštį.

Jeigu negalima, tai užrašyti atitinkamą tekstą.

Algoritmas. Sprendžiant šią užduotį galime įsivaizduoti, kad trys lazdos geometrijoje atitinka tris atitinkamo ilgio atkarpas a, b ir c. Prisiminkite, kad ne visuomet iš trijų atkarpų galima sudėti trikampį. Būtent, negalima sudėti trikampio, jeigu yra bent viena pora atkarpų, kurios ilgių suma yra mažesnė arba lygi už likusiąją trečiąją atkarpą. Arba galima pasakyti ir taip, kad iš trijų atkarpų trikampį galima sudėti tuomet, kai visų galimų (arba bet kurių) atkarpų porų ilgių suma yra didesnė už trečiąją atkarpą, t. y.

(a + b > c) ir (a + c > b) ir (c + b > a).\*/

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

int a, b, c, n;

cout << "kiek lazdu rinkiniu yra ? " << endl;

cin >> n;

for (int i; i <= n; i++)

{

cout << "įveskite 3-ju lazdu ilgius: " << endl;

cin >> a >> b >> c;

cout << "Lazdos: " << setw(2) << fixed << a << " "

<< setw(2) << fixed << b << " "

<< setw(2) << fixed << c << endl;

if ((a + b > c)&& (a + c > b) && (b + c > a)) //ar trikampis?

if ((a == b) && (b == c)) // ar lygiakrastis?

cout << "galima sudaryti lygiakrasti trikampį " << endl;

else if ((a == b) || (b == c) || (a == c)) //ar lygiasonis

cout << "galima sudaryti lygiašonį trikampį " << endl;

else cout << "galima sudaryti įvairiakraštį trikampį" <<endl;

else cout << "trikampio sudaryti negalima" << endl;

}

return 0;

}

/\* Užduotis. Tarkime, kad turime sveikąjį skaičių sk, sudarytą iš n (n – lyginis skaičius) skaitmenų. Jeigu skaičius sk turi 2 daugiklius, kiekvienas iš kurių yra sudarytas iš n/2 skaitmenų ir sudarytas iš skaičiaus sk skaitmenų (visi skaitmenys naudojami tik po 1 kartą), tai jis yra vadinamas vampyro skaičiumi, o iš jo skaitmenų sudaryti daugikliai –vampyro iltimis. Parašykite programą, kuri surastų ir ekrane atspausdintų visus keturženklius vampyro skaičius ir jų iltis.

Algoritmas. Prieš pradedant nagrinėti šią užduotį pateiksime keletą keturženklių vampyro skaičių ir jo „ilčių” pavyzdžių:

1260 = 21 \* 60

1395 = 15 \* 93

1827 = 21 \* 87

Kaip pastebite, šie keturženkliai skaičiai yra išskaidyti į dviejų dviženklių skaičių sandaugą. Be to, šie abu skaičiai turi visus pradinio skaičiaus skaitmenis.

Norint pradėti spręsti šią užduotį pirmiausia reikia mokėti atskirti skaičiaus skaitmenis. Tai padaryti galima naudojant sveikųjų skaičių dalybos operatorius / ir %. Primename, kad pirmasis iš jų leidžia gauti sveikąją dalį, o antrasis – dalybos liekaną. Pavyzdžiui, turime keturženklį skaičių sk = 1234. Pirmąjį skaičiaus skaitmenį pažymėkime a, antrąjį – b, trečiąjį – c, ketvirtąjį – d simboliais, t. y. sk = abcd. Dabar parašykime formules, pagal kurias galima rasti šiuos skaitmenis:

a = sk / 1000

b = (sk / 100) % 10

c = (sk / 10) % 10

d = sk % 10

Perkelkite tai, kas buvo užrašyta, į programą.\*/

#include <iostream>

using namespace std;

int main()

{

int sk, n, a, b, c, d;

for (int sk = 1000; sk <= 9999; sk ++)

{

a = sk / 1000;

b = (sk / 100) % 10;

c = (sk / 10) % 10;

d = sk % 10;

cout << "skaičių poros " << a \* 10 + b << " " << c \* 10 + d << endl;

}

return 0;

}

/\* Užduotis. Tarkime, kad turime sveikąjį skaičių sk, sudarytą iš n (n – lyginis skaičius) skaitmenų. Jeigu skaičius sk turi 2 daugiklius, kiekvienas iš kurių yra sudarytas iš n/2 skaitmenų ir sudarytas iš skaičiaus sk skaitmenų (visi skaitmenys naudojami tik po 1 kartą), tai jis yra vadinamas vampyro skaičiumi, o iš jo skaitmenų sudaryti daugikliai –vampyro iltimis. Parašykite programą, kuri surastų ir ekrane atspausdintų visus keturženklius vampyro skaičius ir jų iltis.

Algoritmas. Prieš pradedant nagrinėti šią užduotį pateiksime keletą keturženklių vampyro skaičių ir jo „ilčių” pavyzdžių:

1260 = 21 \* 60

1395 = 15 \* 93

1827 = 21 \* 87

Kaip pastebite, šie keturženkliai skaičiai yra išskaidyti į dviejų dviženklių skaičių sandaugą. Be to, šie abu skaičiai turi visus pradinio skaičiaus skaitmenis.

Norint pradėti spręsti šią užduotį pirmiausia reikia mokėti atskirti skaičiaus skaitmenis. Tai padaryti galima naudojant sveikųjų skaičių dalybos operatorius / ir %. Primename, kad pirmasis iš jų leidžia gauti sveikąją dalį, o antrasis – dalybos liekaną. Pavyzdžiui, turime keturženklį skaičių sk = 1234. Pirmąjį skaičiaus skaitmenį pažymėkime a, antrąjį – b, trečiąjį – c, ketvirtąjį – d simboliais, t. y. sk = abcd. Dabar parašykime formules, pagal kurias galima rasti šiuos skaitmenis:

a = sk / 1000

b = (sk / 100) % 10

c = (sk / 10) % 10

d = sk % 10

Perkelkite tai, kas buvo užrašyta, į programą.\*/

#include <iostream>

using namespace std;

int main()

{

int sk, n, a, b, c, d;

for (int sk = 2002; sk <= 2009; sk ++)

{

a = sk / 1000;

b = (sk / 100) % 10;

c = (sk / 10) % 10;

d = sk % 10;

cout << "sukeisti skaiciai: " << d << " " << b << " " << c << " " << a << endl;

}

return 0;

}

/\* Užduotis. Tarkime, kad turime sveikąjį skaičių sk, sudarytą iš n (n – lyginis skaičius) skaitmenų. Jeigu skaičius sk turi 2 daugiklius, kiekvienas iš kurių yra sudarytas iš n/2 skaitmenų ir sudarytas iš skaičiaus sk skaitmenų (visi skaitmenys naudojami tik po 1 kartą), tai jis yra vadinamas vampyro skaičiumi, o iš jo skaitmenų sudaryti daugikliai –vampyro iltimis. Parašykite programą, kuri surastų ir ekrane atspausdintų visus keturženklius vampyro skaičius ir jų iltis.

Algoritmas. Prieš pradedant nagrinėti šią užduotį pateiksime keletą keturženklių vampyro skaičių ir jo „ilčių” pavyzdžių:

1260 = 21 \* 60

1395 = 15 \* 93

1827 = 21 \* 87

Kaip pastebite, šie keturženkliai skaičiai yra išskaidyti į dviejų dviženklių skaičių sandaugą. Be to, šie abu skaičiai turi visus pradinio skaičiaus skaitmenis.

Norint pradėti spręsti šią užduotį pirmiausia reikia mokėti atskirti skaičiaus skaitmenis. Tai padaryti galima naudojant sveikųjų skaičių dalybos operatorius / ir %. Primename, kad pirmasis iš jų leidžia gauti sveikąją dalį, o antrasis – dalybos liekaną. Pavyzdžiui, turime keturženklį skaičių sk = 1234. Pirmąjį skaičiaus skaitmenį pažymėkime a, antrąjį – b, trečiąjį – c, ketvirtąjį – d simboliais, t. y. sk = abcd. Dabar parašykime formules, pagal kurias galima rasti šiuos skaitmenis:

a = sk / 1000

b = (sk / 100) % 10

c = (sk / 10) % 10

d = sk % 10

Perkelkite tai, kas buvo užrašyta, į programą.

kiek yra lyginiu ketruezenkliu skaiciu intervale nuo 3001 iki 4005\*/

#include <iostream>

using namespace std;

int main()

{

int sk, n, a, b, c, d;

for (int sk = 3001; sk <= 4005; sk ++)

{

if (sk % 2 == 0)

a = sk / 1000;

b = (sk / 100) % 10;

c = (sk / 10) % 10;

d = sk % 10;

cout << "visi keturzenkliai lyginiai skaiciai: " << a << " " << b << " " << c << " " << d << endl;

}

return 0;

}

/\* Turguje ant prekystalio guli n arbūzų, kurių svoriai žinomi. Reikia sudaryti programą sunkiausiam arbūzui nustatyti.\*/

#include <iostream>

using namespace std;

int main()

{

int n, i, dn;

double d, sv;

cout << "arbuzu skaicius: ";

cin >> n;

cout << "iveskite arbuzu svori: ";

cin >> sv;

d = 0;

dn = 0;

for (i = 1; i<=n; i++)

{

cout <<"Įveskite "<< i <<" arbuzo svori: "<< endl;

cin >> sv;

if (sv > d)

{

d = sv;

dn = i;

}

}

cout << "sunkiausias arbuzas yra: " << dn;

return 0;

}

/\* Turguje ant prekystalio guli n arbūzų, kurių svoriai žinomi. Reikia sudaryti programą sunkiausiam arbūzui nustatyti.\*/

#include <iostream>

#include <fstream>

using namespace std;

int main()

{

int n, i, dn;

double d, sv;

ifstream fd ("duom.txt");

ofstream fr ("rez.txt");

fd >> n;

fd >> sv;

d = sv; dn = 0;

for (i = 2; i<=n; i++)

{

fd >> sv;

if (sv > d)

{

d = sv;

dn = i;

}

}

fr << "sunkiausias arbuzas yra: " << dn;

fr.close();

fr.close();

return 0;

}

III grupė

1. Parašykite algoritmą, kuris suskaičiuotų, keli natūralieji lyginiai skaičiai nuo 1 iki n dalijasi iš penkių.

2. Parašykite algoritmą, kuris rastų klaviatūra renkamų k skaičių sekos nelyginių skaičių sumą.

3. Parašykite algoritmą eilutės sumai skaičiuoti:

b/(a-1) + b/(a-2) +...+ b/(a-b)

/\*1. Parašykite algoritmą, kuris suskaičiuotų, keli natūralieji yginliai skaičiai nuo 1 iki n dalijasi iš penkių.\*/

/\*1. Parašykite algoritmą, kuris suskaičiuotų, keli natūralieji yginliai skaičiai nuo 1 iki n dalijasi iš penkių.\*/

#include <iostream>

using namespace std;

int main()

{

int n, sk, i;

cout << "įveskite didžiausią skaičių: ";

cin >> n;

sk = 0;

for (i = 1; i<=n;i++)

{

if ((i % 2 == 0) && (i%5==0)){

sk++;

}

}

cout << "lyginiu skaičių kurie dalijasi is 5, skaicius yra: " << sk;

return 0;

}

/\*2. Parašykite algoritmą, kuris rastų klaviatūra renkamų k skaičių sekos nelyginių skaičių sumą.\*/

#include <iostream>

using namespace std;

int main()

{

int sk, n;

cout << "įveskite betkokių skaičių: ";

cin >> n;

sk = 0;

for (int i = 1; i<=n;i++)

if (i % 2!= 0)sk += i;

cout << sk;

return 0;

}

/\*3. Parašykite algoritmą eilutės sumai skaičiuoti:

b/(a-1) + b/(a-2) +...+ b/(a-b)\*/

#include <iomanip>

#include <iostream>

using namespace std;

int main()

{ double a, b, h;

cout << "Įveskite skaičius: " << endl;

cin >> a;

cin >> b;

h=0;

for (int i=1; i <= b ;i++)

{

h += b/(a - i);

}

cout << h;

return 0;

}