Обчислювальна геометрія

Омелян П.П.
Гімназія №4, м.Луцьк

Луцьк - 2007

ЗМІСТ

ПЕРЕДМОВА

«Обчислювальна геометрія – це розділ інформатики, що вивчає алгоритми розв'язування геометричних задач. Такі задачі виникають в комп'ютерній графіці, в проектуванні інтегральних схем, технічних пристроїв та іншого. Вхідними даними в такого роду задачах може бути множина точок, набір відрізків, багатокутників і т.д. Результатом може бути відповідь на будь-яке запитання (типу «перетинаються дані прямі, чи ні»), або деякий геометричний об'єкт (наприклад, найменший опуклий багатокутник, який містить задані точки)».

Заняття навіть з математично добре підготовленими учнями старших класів показали, що при розв'язанні геометричних задач в них виникають великі труднощі. Задачі заводять їх в глухий кут, або вибраний «прямий» спосіб розв'язування настільки складний, що довести його до логічного завершення без помилок, учні не можуть. Аналіз результатів розв'язування «геометричних» задач на олімпіадах з інформатики показує такі ж результати. Дану ситуацію можна виправити. Ціль цієї розробки – показати різні підходи до розв'язування геометричних задач на площині, які дозволять дуже швидко і максимально просто отримати розв'язки більшості елементарних підзадач.

Вектори і координати

Щоб використовувати методи обчислювальної геометрії, необхідно геометричні зображення перевести на мову чисел. Будемо рахувати, що на площині задана декартова система координат (СК). Взагалі прийнято вибирати координатні осі так, щоб поворот на кут $\frac{\pi}{2}$, при якому вісь Ох накладається на Оу, здійснювався проти годинникової стрілки. Така СК називається правою. Надалі будемо говорити, що наша СК права. В такій СК напрямок повороту проти годинникової стрілки називають додатнім.

Тепер геометричні об'єкти отримають аналітичні вирази. Так, щоб задати відрізок, достатньо вказати координати його кінців. Пряму можна задати, вказавши дві її точки, або координати однієї її точки і вектор, який характеризує напрямок даної прямої. Взагалі при розв'язуванні задач основним інструментом для нас буде вектор. Нагадаємо для цього деякі відомості про нього.

Відрізок АВ, в якому т. А вважається початком, а т. В – кінцем, називається вектором АВ, і позначається $\overrightarrow{AB}$, або жирним курсивом латинською буквою, ***a***. Для позначення довжини вектора (тобто довжини відповідного відрізка) будемо використовувати символ модуля (наприклад, $|a|$). Два вектора називаються рівними, якщо вони накладаються паралельним перенесенням.

Нехай точки А і В мають координати (x_1, y_1) і (x_2, y_2) відповідно. Координатами вектора $\overrightarrow{AB}$ називається пара чисел (x_2-x_1, y_2-y_1). І навпаки, якщо вектор має координати (x, y) і бере початок з точки (x_1, y_1), то легко обчислити координати його кінця: x_2=x_1+x, y_2=y_1+y. Довжина вектора $\overrightarrow{AB}$ за теоремою Піфагора рівна $\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$. Рівність двох векторів ***a***$=(a_x, a_y)$ і ***b***$=(b_x, b_y)$ еквівалентна рівності їх відповідних координат: a_x=b_x, a_y=b_y.

Вектори можна додавати і множити на скаляр. Сума векторів шукається за правилом трикутника, або за правилом паралелограма (мал. 1).

Мал. 1

Під різницею векторів ***a*** і ***b*** розуміють суму вектора ***a*** з вектором, протилежним вектору ***b*** (тобто протилежно направленим і співпадаючим з ним по довжині). При множені вектора ***a*** на скаляр t отримаємо вектор, який має довжину $|t| \cdot |a|$, його напрямок співпадає з напрямком ***a***, якщо t>0, і протилежний йому, якщо t<0. Це дозволяє нам вивести співвідношення колінеарних векторів(тобто однаково напрямлених, або протилежно напрямлених), підрозумівають під цим коефіцієнт їх пропорційності. За допомогою даного співвідношення зручно описувати порядок розміщення точок на прямій. Наприклад, умова $\frac{\overrightarrow{AB}}{\overrightarrow{AC}} < 0$ означає, що точки A, B і C лежать на одній прямій, причому т. A лежить між B і C. Відмітимо ще, що вектор, співнапрямлений з даним вектором ***a*** і має задану довжину *l* можна представити таким чином $\frac{l}{|a|}a$. Надалі ми неодноразово будемо цим користуватися. В координатах перераховані операції над векторами записуються так:

якщо ***a***$=(a_x, a_y)$ і ***b***$=(b_x, b_y)$,

то ***a***+***b***=(a_x+b_x, a_y+b_y), ***a***-***b***=(a_x-b_x, a_y-b_y) і t****a***=(t*a_x, t*a_y).

Скалярний добуток двох ненульових векторів ***a***$=(a_x, a_y)$ і ***b***$=(b_x, b_y)$ називають число $(a,b) = |a| \cdot |b| \cdot \cos\varphi$, де φ - кут між даними векторами.

В координатах це обчислюється таким чином:

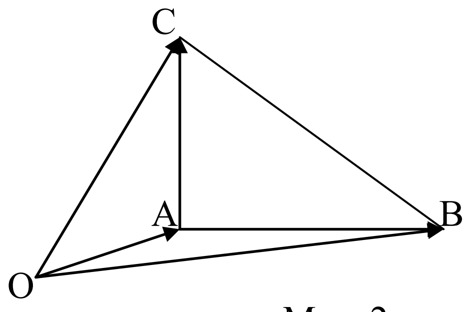
$(a,b)=x_1x_2+y_1y_2$.

Як видно з формули, скалярний добуток можна використовувати для знаходження кута між векторами. Взагалі, два ненульових вектора перпендикулярні тоді і тільки тоді, коли їх скалярний добуток рівний нулю ($\cos\frac{\pi}{2}=0$). Так як $\cos\varphi$ додатній для гострих кутів і від'ємний для тупих, кут між векторами гострий (тупий) в тому випадку, коли їх скалярний добуток додатній (від'ємний).

Кут між векторами

Нехай ***a*** і ***b*** – два ненульових вектори, відкладені з однієї точки. В шкільній програмі з геометрії під кутом між векторами розуміють менший з двох кутів між променями, на яких лежать вектори ***a*** і ***b***. Значення такого кута завжди знаходиться на проміжку $[0;\pi]$.

Для обчислень найбільш зручнішим є поняття орієнтованого кута, тобто кута, що враховує взаємне розміщення векторів. Значення орієнтованого кута за абсолютною величиною рівне звичайному куту між векторами. Орієнтований кут між векторами ***a*** і ***b*** додатній, якщо поворот від вектора ***a*** до вектора ***b***, здійснюється в додатному напрямку (в нашій СК проти годинникової стрілки), і від'ємний в протилежному випадку. Таким чином, величина орієнтованого кута залежить від порядку перерахунку векторів і може набувати значення на інтервалі $(-\pi;\pi]$. На мал. 2 орієнтування кута між векторами $\overrightarrow{OA}$ і $\overrightarrow{OB}$ і між векторами $\overrightarrow{OB}$ і $\overrightarrow{OA}$ рівне по модулю, але перший з них від'ємний, а другий додатній. Для будь-яких векторів $\overrightarrow{OA}$, $\overrightarrow{OB}$ і $\overrightarrow{OC}$ легко обчислити величину орієнтованого кута AOB, знаючи величини кутів AOC і COB: вона рівна їх сумі з врахуванням знаків. Наприклад, при такому розміщенні векторів, як на мал. 2, кут AOC буде в сумі зі знаком плюс, а кут COB – з мінусом. Може статися, що при сумі рівних двох додатних (двох від'ємних) кутів результат стане більший за π по модулю. Тоді, щоб отримати правильне значення кута, потрібно відняти (додати) 2π. При цьому нам непотрібно розглядати різні випадки взаємного розміщення векторів. В цьому і є перевага використання орієнтованих кутів.



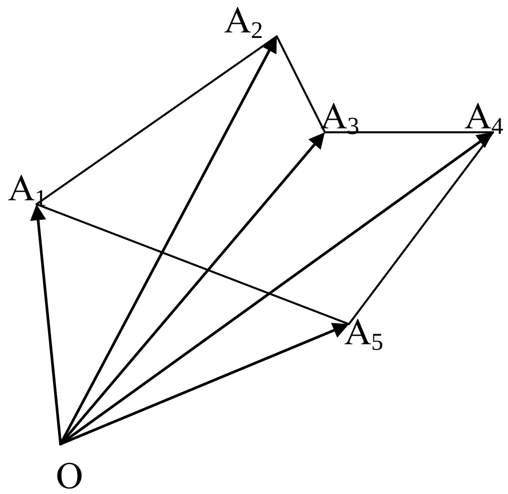
Мал. 2

Як, знаючи координати векторів, знайти кут між ними. Звичайно, з формули скалярного добутку: $\cos\varphi = \frac{(a,b)}{|a| \cdot |b|}$. Але при цьому отримаємо значення неорієнтованого кута і частина інформації (можливо важливої) буде нами втрачена. Крім цього, використання даної формули для програмування не завжди зручне. Наприклад, в мові програмування Паскаль, як і в багатьох інших мовах програмування з обернених тригонометричних функцій реалізована тільки функція $arctg\varphi$. Покажемо, як можна знайти кут інакше, після того, як познайомимось з орієнтованою площею.

Орієнтована площа

Орієнтована площа трикутника – це його звичайна площа наділена знаком. Знак в орієнтованій площі трикутника ABC такий ж, як і в орієнтованого кута між векторами $\overrightarrow{AB}$ і $\overrightarrow{AC}$. Тобто її знак залежить від порядку перерахунку вершин. На мал. 2 трикутник ABC – прямокутний. Його орієнтована площа рівна $\frac{\left|\overrightarrow{AB}\right| \cdot \left|\overrightarrow{AC}\right|}{2}$ (вона більша 0, так як пара $\overrightarrow{AB}$ і $\overrightarrow{AC}$ орієнтована додатньо). Дану величину можна обчислити іншим способом. Нехай O – довільна точка площини. На нашому малюнку площу трикутника ABC отримуємо, якщо від площі трикутника OBC відняти площі OAB і OCA. Таким чином, потрібно просто додати орієнтовані площі трикутників OAB, OBC і OCA. Це правило правильне при будь-якому розміщенні т. O.

Так само для обчислення площі будь-якого багатокутника $A_1A_2\ldots A_n$ потрібно додати площі трикутників OA_1A_2, OA_2A_3, …, OA_nA_1 (мал. 3). В сумі отримаємо площу багатокутника, взяту із знаком плюс, якщо при обході ламаної $A_1A_2\ldots A_n$ внутрішня частина багатокутника знаходиться з ліва, і із знаком мінус, якщо вона знаходиться справа. Вона і називається орієнтованою площею багатокутника $A_1A_2\ldots A_n$. Таким чином, для правої СК орієнтована площа виявиться додатною при обході границі багатокутника проти годинникової стрілки.



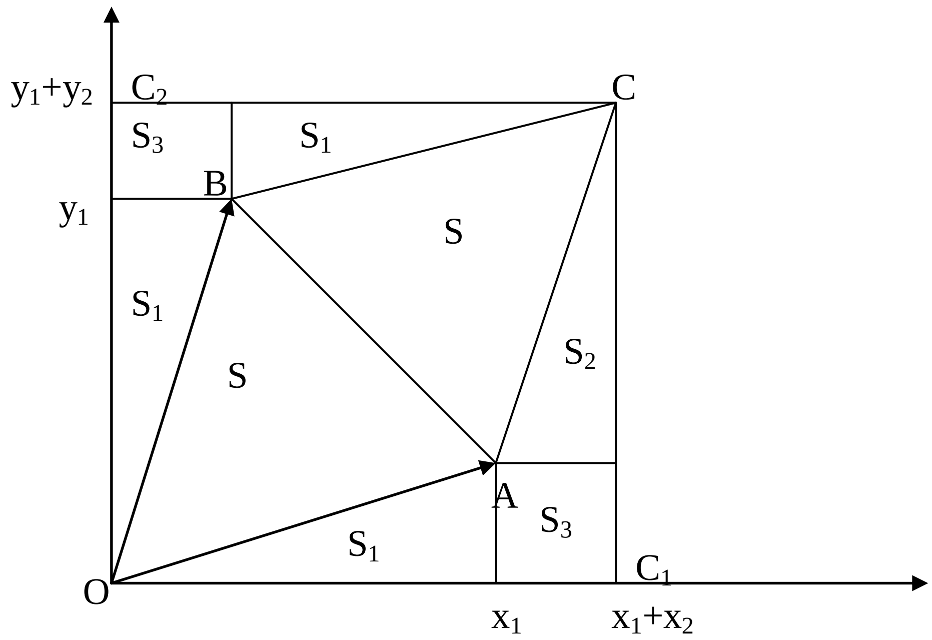
Мал. 3

Отже, обчислення площі багатокутника звелося до знаходження орієнтованої площі трикутника. Розглянемо, як записати її за допомогою координат. Нехай S – орієнтована площа трикутника побудованого на векторах $\boldsymbol{a}=(x_1,y_1)$ і $\boldsymbol{b}=(x_2,y_2)$. Обчислимо її для даного розміщення векторів (мал. 4). Величина S тут додатна (пара векторів $\boldsymbol{a}=\overrightarrow{OA}$ і $\boldsymbol{b}=\overrightarrow{OB}$ додатньо орієнтовані). Добудуємо наш трикутник до паралелограма OACB з площею 2S (тут $\overrightarrow{OC}=\overrightarrow{OA}+\overrightarrow{OB}$). Тоді площа прямокутника OC_1CC_2 становитиме

$$|OC_1|\cdot|OC_2|=(x_1+x_2)(y_1+y_2)=2S+2S_1+2S_2+2S_3=2S+x_1y_1+x_2y_2+2x_2y_1$$

(тут S_1, S_2, S_3 – звичайні не орієнтовані площі). Розкриємо дужки в лівій частині рівності і виразимо 2S, отримаємо

$$2S=x_1y_2-x_2y_1 \qquad \textbf{(1)}$$



Мал. 4

Легко переконатись, що при інших варіантах розміщення векторів формула (1) також залишається правильною. Таким чином, орієнтована площа паралелограма, побудована на векторах ***a***=(x_1,y_1) і ***b***=(x_2,y_2), дорівнює x_1y_2-x_2y_1.

Величина x_1y_2-x_2y_1 називається косим (або псевдоскалярним) добутком векторів ***a*** і ***b***. Для «косого» добутку ми будемо використовувати позначення [***a,b***] . Ця назва пов'язана із властивістю косої симетрії: [***a***,***b***]=-[***a***,***b***] (Дане позначення використовується для векторного добутку, але на відміну від векторного добутку косий добуток – це скаляр). Так, як неорієнтована площа паралелограма, побудованого на векторах ***a*** і ***b***, рівна $|a|\cdot|b|\cdot\sin\varphi$, а знак $\sin\varphi$ співпадає із знаком орієнтованого кута φ, тоді [***a***,***b***]= $|a|\cdot|b|\cdot\sin\varphi$. Величина [***a***,***b***] більша нуля, якщо пара векторів ***a*** і ***b*** додатньо орієнтована і менше нуля в протилежному випадку. Косий добуток ненульових векторів рівний нулю тоді і тільки тоді, коли вони колінеарні ($\sin 0 = \sin\pi = 0$).

Тепер, як і обіцяли, знайдемо в координатах кут між двома векторами. Нехай φ орієнтований кут між векторами ***a***=(x_1,y_1) і ***b***=(x_2,y_2). З скалярного добутку і косого добутку випливає формула $tg\varphi = \dfrac{[a,b]}{(a,b)} = \dfrac{x_1y_2 - x_2y_1}{x_1x_2 + y_1y_2}$. Знаючи тангенс кута між векторами, ми легко знайдемо кут між прямими на яких лежать ***a*** і ***b***: він дорівнює $\left| arctg\,\frac{[a,b]}{(a,b)} \right|$. Щоб отримати сам кут між векторами, залишилось з'ясувати гострий він чи тупий. Це ми визначимо зі знаку скалярного добутку. Врахуємо ще, що знак орієнтованого кута співпадає із знаком косого добутку. Тоді отримаємо:

$\varphi = \frac{\pi}{2}$, якщо (a,b)=0, [a,b]>0;

$\varphi = -\frac{\pi}{2}$, якщо (a,b)=0, [a,b]<0;

$\varphi = arctg\,\frac{[a,b]}{(a,b)}$, якщо (a,b)>0;

$\varphi = arctg\,\frac{[a,b]}{(a,b)} + \pi$, якщо (a,b)<0, [a,b]>=0;

$\varphi = arctg\,\frac{[a,b]}{(a,b)} - \pi$, якщо (a,b)<0, [a,b]<0. **(2)**

Величина звичайного кута рівна модулю значення орієнтованого кута.

Відмітимо, що всі відомості про орієнтовані кути і площі відносились до правої СК. Може таке статись, що для конкретної задачі краще використовувати ліву СК (вісь абсцис направлена в право, вісь ординат – вниз). При такому розміщені осей додатнім буде поворот за годинниковою стрілкою. З цією поправкою все вище сказане застосовується і до лівої СК.

РІВНЯННЯ ЛІНІЙ

1.1. Рівняння прямої, яка проходить через дві різні точки, які задані своїми координатами.

Нехай на прямій задані дві точки P_1 з координатами (x_1,y_1) і P_2 з координатами (x_2,y_2). Відповідно вектор з початком т. P_1 і кінцем в т. P_2 має координати (x_2-x_1,y_2-y_1). Якщо $P(x,y)$ – довільна точка нашої прямої, то координати вектора $\overrightarrow{P_1P}$ становлять $(x-x_1,y-y_1)$. За допомогою косого добутку умова колінеарності векторів $\overrightarrow{P_1P}$ і $\overrightarrow{P_1P_2}$ можна виразити таким чином: $[\overrightarrow{P_1P},\overrightarrow{P_1P_2}]=0$, тобто

$$(x-x_1)(y_2-y_1)-(y-y_1)(x_2-x_1)=0 \quad \textbf{(3)}$$

або

$$(y_2-y_1)x+(x_1-x_2)y+x_1(y_1-y_2)+y_1(x_2-x_1)=0$$

Остатню рівність запишемо так:

$$ax+by+c=0 \quad \textbf{(4)}$$

де $a=y_2-y_1$,

$b=x_1-x_2$,

$c= x_1(y_1-y_2)+y_1(x_2-x_1)$.

Отже, будь-яку пряму можна задати рівнянням виду (4). В наступному пункті покажемо, що і навпаки, при будь-яких значеннях коефіцієнтів (крім a=b=0) рівняння такого виду задає на площині деяку пряму.

Зауважимо, що при програмуванні формулу (3) не можна використовувати у вигляді відношення: $\frac{x_2-x_1}{x-x_1}=\frac{y_2-y_1}{y-y_2}$, так як, по-перше, навіть якщо всі координати заданих точок цілі числа, помилки дійсної арифметики при дії ділення не дозволяє перевірити за допомогою даного відношення, належність деякої точки даній прямій, а, по-друге, програма буде зупинена при діленні на нуль.

Рівняння прямої можна записати і в параметричному вигляді. Будь-який вектор, з початком в т. P_1 і кінцем в довільній т. P(x,y), який лежить на даній прямі, можна отримати з $\overrightarrow{P_1P_2}$, шляхом множення вектора на деяке дійсне число t. Тоді для кожної з координат справедлива рівність

$$(x-x_1)=t(x_2-x_1) \text{ і } (y-y_1)=t(y_2-y_1)$$

Знайшовши звідси x і y, отримаємо систему параметричних рівнянь, яка задовольняє кожну точку нашої прямої.

$$\begin{cases} x = x_1 + t(x_2 - x_1), \\ y = y_1 + t(y_2 - y_1). \end{cases} \quad \textbf{(5)}$$

І навпаки, якщо координати (x,y) точки P задовольняють дані співвідношення, вектор $\overrightarrow{P_1P}$ колінеарний вектору $\overrightarrow{P_1P_2}$ і це означає, що т. P лежить на прямі P_1P_2.

Ця система, але із введеними обмеженнями значення t, буде задавати і відрізок P_1P_2, і промінь P_1P_2. Для відрізка $t \in [0,1]$ (тобто x змінюється в діапазоні $[x_1,x_2]$, а y – в діапазоні $[y_1,y_2]$), а для променя - $t \in [0,\infty)$.

1.2. Рівняння прямої, заданою однією з її точок і вектором нормалі до неї.

Нехай задана точка P_o прямої має координати (x_o,y_o), а деякий вектор нормалі ***n*** до неї (тобто вектор ортогональний до нашої прямої) – координатами (a,b). Якщо P(x,y) довільна точка нашої прямої, тоді координати вектора $\overrightarrow{P_oP}$ рівні $(x\text{-}x_o,\ y\text{-}y_o)$. Скалярний добуток ортогональних векторів $(n,\overrightarrow{P_oP})$ можна виразити так:

$$(n,\overrightarrow{P_oP}) = a(x - x_o) + b(y - y_o) = 0 \qquad \textbf{(6)}$$

Очевидно, що рівняння (6) також легко звести до вигляду (4). Тоді видно, що коефіцієнти a, b у рівнянні (4) будуть координатами одного із векторів нормалі до нашої прямої. Звідси, випливає, що при будь-яких значеннях коефіцієнтів a, b і c (крім a=b=0) рівняння (4) задає пряму. Це буде пряма, перпендикулярна вектору (a, b) і яка проходить через точку, координати якої задовольняють (4). При $a \neq 0$ такою точкою буде, наприклад, точка $(-\frac{c}{a},0)$, при a=0 – точка $(0,-\frac{c}{b})$.

Не дивлячись на те, що на перший погляд умова задачі, може здатись надуманою, саме за допомогою її ми отримаємо хороший інструмент для розгляду цілого класу задач. В чому і переконаємось.

1.3. Рівняння прямої, перпендикулярної даній, і що проходить через задану точку.

Нехай задана точка P_o шуканої прямої має координати (x_o,y_o). Якщо P(x,y) – довільна точка на цій прямій, то координати вектора $\overrightarrow{P_oP}$ дорівнюють $(x\text{-}x_o,y\text{-}y_o)$. Цей вектор перпендикулярний вектору $\overrightarrow{P_1P_2}$, де $P_1(x_1,y_1)$ і $P_2(x_2,y_2)$ – точки на даній прямі. Тоді скалярний добуток ортогональних векторів $(\overrightarrow{P_1P_2},\overrightarrow{P_oP})$ можна виразити так:

$$(x_2-x_1)(x-x_o)++(y_2-y_1)(y-y_o)=0 \qquad \textbf{(7)}$$

або

$$(x_2-x_1)x+(y_2-y_1)y+(x_2-x_1)x_o+(y_2-y_1)y_o=0.$$

Якщо наша пряма задана коефіцієнтами a, b і c свого рівняння, то легко побачити, що вектор її нормалі з координатами (a,b) колінеарний вектору $\overrightarrow{P_oP}$.

Тоді, записавши косий добуток цих векторів, отримаємо:

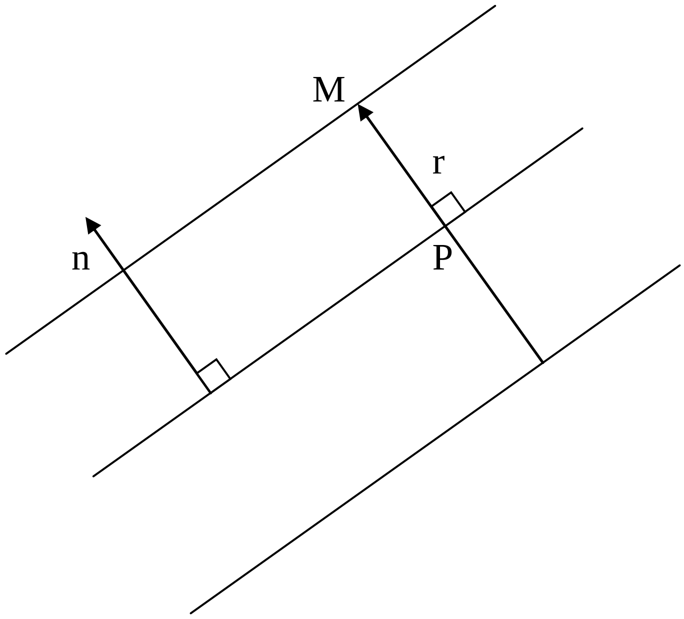
$$b(x-x_0)-a(y-y_0)=0 \qquad \textbf{(8)}$$

1.4. ***Рівняння прямої, що паралельна даній, та знаходиться на відстані r.***

Очевидно, що шуканих прямих дві. Вектор нормалі до даної прямої ортогональний і до кожної з паралельних прямих. Значить коефіцієнти a і b при x і y в рівнянні (4) для паралельних прямих такі самі, як і в заданій прямій. Залишається підібрати значення третього коефіцієнта. Позначимо його для однієї прямої c_1, для другої – c_2. Як вище було зазначено, для визначення цих коефіцієнтів достатньо знайти хоча б по одній точці на кожній із прямих.

Візьмемо довільну точку $P(x_0, y_0)$ на заданій прямій (якщо пряма була задана не двома точками, то точку можна знайти за алгоритмом поданим в кінці п. 1.2). Проведемо через неї пряму, перпендикулярну даній. На паралельній прямі будемо шукати точку перетину ($M(x_1,y_1)$) з цим перпендикуляром (мал. 5). Нам відомий один вектор нормалі ***n***=(a,b). Вектор $\overrightarrow{PM}$ колінеарний йому, а його довжина r. Для визначеності будемо рахувати, що, як і на малюнку, нормаль лежить з того ж боку прямої, що і точка M. Тоді $\overrightarrow{PM} = n \cdot \frac{r}{\sqrt{a^2+b^2}}$. Значить координати точки M дорівнюють $x_0 + a \cdot \frac{r}{\sqrt{a^2+b^2}}$, $y_0 + b \cdot \frac{r}{\sqrt{a^2+b^2}}$. Підставимо їх у формулу (6), отримаємо $c_1 = -ax_0 - by_0 - r\sqrt{a^2+b^2}$.

Рівняння однієї із прямих отримане. В якості вектора нормалі для другої прямої можна взяти вектор – $\overrightarrow{PM}$. В даному випадку отримаємо $c_2 = -ax_0 - by_0 + r\sqrt{a^2 + b^2}$.

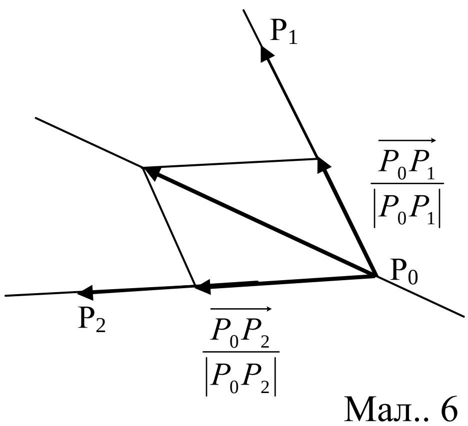


Мал. 5

1.5. Рівняння бісектриси кута.

Нехай вектор $\overrightarrow{P_0P_1}(x_1, y_1)$ і $\overrightarrow{P_0P_2}(x_2, y_2)$ беруть початок в точці $P_0(x_0, y_0)$. Знайдемо рівняння бісектриси кута $P_1P_0P_2$. Якщо ми поділимо кожен з векторів $\overrightarrow{P_0P_1}$ і $\overrightarrow{P_0P_2}$ на його довжину, отримаємо при цьому вектори одиничної довжини. Вектор їх суми буде лежати на бісектрисі кута між ними (мал. 6). Координатами даного вектора будуть:

$$\frac{x_1}{\sqrt{x_1^2 + y_1^2}} + \frac{x_2}{\sqrt{x_2^2 + y_2^2}}; \frac{y_1}{\sqrt{x_1^2 + y_1^2}} + \frac{y_2}{\sqrt{x_2^2 + y_2^2}}.$$



Мал.. 6

З умови колінеарності вектора $\overrightarrow{P_0P}$, де P(x,y) – довільна точка на шуканій прямій, отримаємо рівняння бісектриси:

$$\left(\frac{y_1}{\sqrt{x_1^2+y_1^2}}+\frac{y_2}{\sqrt{x_2^2+y_2^2}}\right)(x-x_0)-\left(\frac{x_1}{\sqrt{x_1^2+y_1^2}}+\frac{x_2}{\sqrt{x_2^2+y_2^2}}\right)(y-y_0) \quad \textbf{(9)}$$

При необхідності з (9) легко отримати коефіцієнти a, b і c для запису рівняння знайденої прямої у вигляді (4).

1.6. ***Рівняння кола.***

За означенням кола з центром в т. $M_0(x_0,y_0)$ і радіусом r, точка M(x,y) лежить на колі тоді і тільки тоді, коли відстань між точками M_0 і M рівна r. Використавши формулу для обчислення квадрату відстані між двома точками, ми прийдемо до наступного рівняння кола:

$$(x-x_0)^2+(y-y_0)^2=r^2 \qquad \textbf{(10)}$$

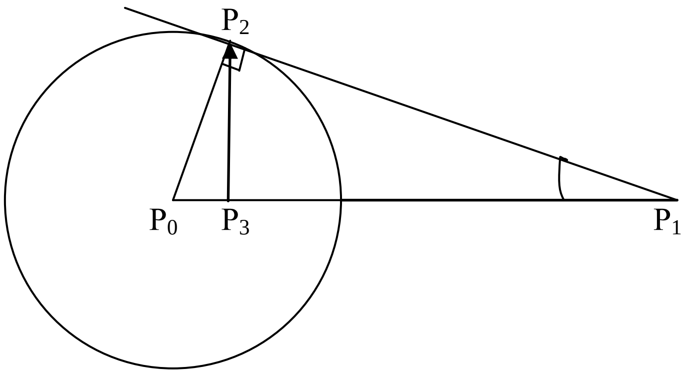
На практиці виявляється корисно знати і параметричне рівняння кола. Розглянемо перше коло з центром в початку координат. Якщо позначити t – кут між радіус-вектором $\overrightarrow{OM}$ (M(x,y) – довільна точка, що лежить на колі) і віссю Ox, який вимірюється проти годинникової стрілки, то очевидно, що x=rcost, y=rsint. Отже, для довільного кола параметричні рівняння будуть мати такий вигляд:

$$x=x_0+r\cos t$$
$$y=y_0+r\sin t \qquad \textbf{(11)}$$

Покажемо ще, як можна визначити довжину l найменшої дуги кола, якщо кінці дуги (x_1,y_1) і (x_2,y_2). З курсу геометрії відомо, що довжина кола прямо пропорційна куту φ між векторами (x_1-x_0,y_1-y_0) і (x_2-x_0,y_2-y_0): $l=r\varphi$. А як знайти значення даного кута, було вже показано вище (потрібно тільки врахувати, що у випадку пошуку довжини меншої з двох дуг нас цікавить неорієнтований кут в діапазоні $[0;\pi]$).

1.7. Дотична до кола.

Нехай коло з центром у т. $P_0(x_0,y_0)$ і радіусом r. Потрібно знайти рівняння дотичної до кола, яка проходить через т. $P_1(x_1,y_1)$. Тут можливі три випадки. Якщо $\left|\overrightarrow{P_0P_1}\right| = r$, то P_1 лежить на колі. Тоді у шуканій дотичній нам відома точка P_1 і нормаль $\overrightarrow{P_0P_1}$, її рівняння легко записується (див. п. 1.3). У випадку $\left|\overrightarrow{P_0P_1}\right| > r$ точок дотику дві. Позначимо одну з них P_2, ми отримаємо прямокутний трикутник $P_0P_2P_1$ (мал. 7). Можна спробувати знайти шукане рівняння, в «лоб». Якщо (x_2,y_2) – координати точки дотику P_2, то за теоремою Піфагора обчислюється довжина відрізка P_1P_2 ($P_0P_2 = r$, P_0P_1 обчислюється за відомими координатами). Друге співвідношення для координат x_2 і y_2 – з рівняння кола (10). Обидва рівняння квадратні. Розв’язання даної системи представляє собою серйозні труднощі. Спробуємо обійтись без квадратних рівнянь, використавши скалярний і косий добуток векторів.



Мал. 7

Ми будемо шукати $a = x_2 - x_1$ і $b = y_2 - y_1$ вектора $\overrightarrow{P_1P_2}$. Як було зазначено вище, довжини сторін прямокутного трикутника $P_0P_2P_1$ легко знайти. Визначимо скалярний добуток векторів $\overrightarrow{P_1P_2}$ і $\overrightarrow{P_1P_0}$: $(\overrightarrow{P_1P_2}; \overrightarrow{P_1P_0}) = |P_1P_2| \cdot |P_1P_0| \cdot \cos\varphi = |P_1P_2|^2$.

Геометричний зміст косого добутку $[\overrightarrow{P_1P_0}, \overrightarrow{P_1P_2}]$ - подвоєння площі трикутника $P_0P_2P_1$, взятої із знаком плюс для однієї з точок і з мінусом – для другої.

$$[\overrightarrow{P_1P_0}, \overrightarrow{P_1P_2}] = \pm |P_0P_2| \cdot |P_1P_2|.$$

Записуючи ці добутки в координатах отримаємо систему лінійних рівнянь відносно a і b:

$$\begin{cases}(x_0 - x_1)\cdot a + (y_0 - y_1)\cdot b = |P_1P_2| \\ (x_0 - x_1)\cdot b - (y_0 - y_1)\cdot a = \pm|P_0P_2|\cdot|P_1P_2|\end{cases}$$

Таку систему розв'язати вже неважко. Далі за точкою $P_1(x_1,y_1)$ і направляючим вектором $\overrightarrow{P_1P_2} = (a,b)$ записуємо рівняння дотичної. Задача розв'язана. Якщо нам потрібно ще знайти і координати точки дотику, то це можна зробити, використавши координати т. P_1 і знайдені координати $\overrightarrow{P_1P_2}$.

Для розв'язування цієї задачі є метод, в якому не потрібно навіть розв'язувати систему лінійних рівнянь. Проведемо з вершини P_2 прямого кута висоту P_2P_3 (мал. 7). З подібності трикутників $P_1P_2P_0$ і $P_1P_3P_2$ знайдемо довжини відрізків P_1P_3 і P_3P_2:

$$|P_1P_3| = \frac{|P_1P_2|^2}{|P_0P_1|}; \; |P_2P_3| = \frac{|P_1P_2|\cdot|P_0P_2|}{|P_0P_1|}.$$

Тепер поступово знаходимо координати вектора $\overrightarrow{P_1P_3}$, точки $P_3(x_3,y_3)$ і, на кінець використовуючи відомі координати вектора $\boldsymbol{n}=(y_0-y_1,\; x_1-x_0)$, перпендикулярного прямій P_1P_3, координати точки P_2:

$$\overrightarrow{P_1P_3} = \overrightarrow{P_1P_0}\cdot\frac{|P_1P_3|}{|P_1P_0|};$$

$$x_3 = x_1 + (\overrightarrow{P_1P_3})_x,\; y_3 = y_1 + (\overrightarrow{P_1P_3})_y;$$

$$\overrightarrow{P_3P_2} = n\cdot\frac{|P_3P_2|}{|n|};$$

$$x_2 = x_3 + (\overrightarrow{P_3P_2})_x,\; y_2 = y_3 + (\overrightarrow{P_3P_2})_y;$$

ВЗАЄМНЕ РОЗМІЩЕННЯ ТОЧОК І ФІГУР

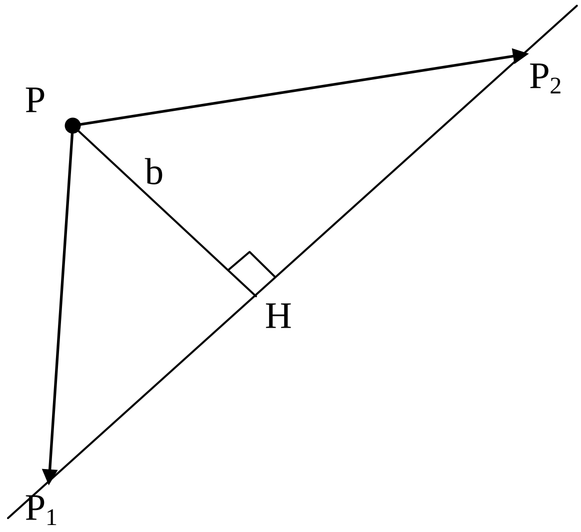
2.1. ***Розміщення точки відносно прямої, променя або відрізка.***

В першу чергу нас цікавить належність даної точки $P(x,y)$ вказаному геометричному об'єкту, рівняння якого нам відоме. Щоб відповісти на це запитання для прямої, достатньо підставити координати заданої точки в рівняння прямої (3) або (4). Рівність нулю значення отриманого виразу (для дійсних координат або коефіцієнтів рівняння перевірку на рівність нулю необхідно здійснювати з урахуванням похибки) означає, що точка належить даній прямі. Якщо значення виразу менше нуля, то точка належить одній півплощині, а більше нуля – іншій. Дійсно рівняння (3) отримаємо за умови, що пряма проходить через точки $P_1(x_1, y_1)$ і $P_2(x_2, y_2)$. Зауважимо, що ліва частина (3) – це значення косого добутку векторів $\overrightarrow{P_1P}$ і $\overrightarrow{P_1P_2}$. Його знак визначає орієнтацію цієї пари векторів, інакше кажучи, належність точки P одній із півплощин (а при рівності нулю – належність прямій).

Якщо пряма задана двома своїми точками, для розв'язування задачі достатньо обчислити значення косого добутку, а рівняння прямої записувати не потрібно.

У випадку перевірки належності точки променю при $\left[\overrightarrow{P_1P}, \overrightarrow{P_1P_2}\right] = 0$ (P_1 – початок променя, P_2 будь-яка точка на промені) корисно знайти і скалярний добуток цих векторів. Якщо він менший нуля, P_1 лежить на прямій між P_2 і P, звідси P не належить променю. Щоб в аналогічній ситуації впевнитись належності точки P відрізку P_1P_2, необхідно обчислити ще і значення скалярного добутку $(\overrightarrow{P_2P_1}, \overrightarrow{P_2P})$. Якщо воно невід'ємне, то точка P лежить на відрізку.

Нехай нам тепер необхідно визначити, на якій відстані знаходиться задана точка до прямої, променя, відрізка. Формулу відстані від точки до прямої отримуємо із співвідношення двох способів обчислення площі трикутника: $2S = b \cdot |P_1P_2| = \left|\left[PP_1, PP_2\right]\right|$ (див. мал. 8). Тобто відстань b від точки P до прямої, заданої координатами точок P_1 і P_2, можна обчислити, як відношення модуля косого добутку векторів $\overrightarrow{PP_1}$ і $\overrightarrow{PP_2}$ до довжини відрізка P_1P_2.



Мал. 8

Для променя або відрізка вказаний спосіб знаходження відстані потрібно дещо відредагувати. Точка H (мал. 8) лежить на промені P_1P_2 в тому випадку, коли скалярний добуток $(\overrightarrow{P_1P_2}, \overrightarrow{P_1P}) \geq 0$. Для відрізка P_1P_2 ясно, потрібно ще і виконання умови $(\overrightarrow{P_2P_1}, \overrightarrow{P_2P}) \geq 0$. Тоді можна застосувати формулу відстані від точки до прямої. В противному випадку відстань до променя (відрізка) буде дорівнювати відстані від точки до початку променя (до найближчого кінця відрізка).

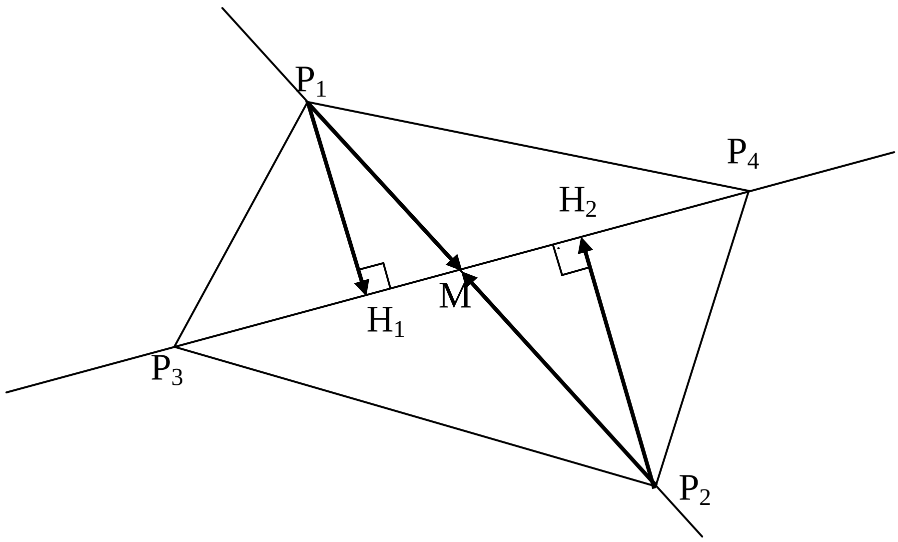
2.2. ***Взаємне розміщення двох точок відносно прямої.***

В багатьох випадках потрібно з'ясувати, по одну чи різні сторони відносно прямої лежать дві точки P_1 і P_2, задані своїми координатами. Виберемо на прямій дві довільні точки, що не співпадають – P_3 і P_4 (ними можуть бути якраз дві точки, що задають нашу пряму). Тоді, якщо косі добутки $[\overrightarrow{P_3P_1}, \overrightarrow{P_3P_4}]$ і $[\overrightarrow{P_3P_2}, \overrightarrow{P_3P_4}]$ мають різні знаки, тобто $[\overrightarrow{P_3P_1}, \overrightarrow{P_3P_4}] \cdot [\overrightarrow{P_3P_2}, \overrightarrow{P_3P_4}] < 0$, то точки лежать по різні сторони прямої, якщо знаки співпадають – то по одну сторону, якщо одне із значень 0, то відповідна точка лежить на прямій. Взагалі ця задача є частиною, більш складної задачі.

2.3. ***Взаємне розміщення двох прямих, або променів і відрізків.***

Легко з'ясувати, чи перетинаються дві прямі чи вони паралельні. Нагадаємо ще раз, що умова колінеарності двох векторів – це рівність нулю їх косого добутку. Якщо прямі задані рівняннями $a_1x+b_1y+c_1=0$ і $a_2x+b_2y+c_2=0$, то зручно перейти до їх нормалей: $n_1=(a_1, b_1)$ і $n_2=(a_2, b_2)$. Тоді умова колінеарності нормалей (а значить, і паралельності прямих): $[\boldsymbol{n_1},\boldsymbol{n_2}]=a_1b_1-a_2b_2=0$. Якщо прямі задані парами точок, то таким ж методом перевіряється колінеарність направляючих векторів. Перевірка наявності перетину прямої і відрізка нами фактично була показана при розв'язуванні задачі 2.2.

Нехай пряма і відрізок перетинаються в одній точці. Знайдемо її. Позначимо кінці відрізка $P_1(x_1, y_1)$ і $P_2(x_2, y_2)$. Нехай $P_3(x_3, y_3)$ і $P_4(x_4, y_4)$ – дві точки на прямій, а M – шукана точка перетину (мал. 9). Опустимо з кінців відрізка перпендикуляри на пряму P_1H_1 і P_2H_2. Трикутники $P_3P_1P_4$ і $P_3P_2P_4$ мають спільні основи. Звідси випливає, що відношення їх площ дорівнює відношенню їх висот $|P_1H_1|$ і $|P_2H_2|$. Але площа (орієнтована) трикутника $P_3P_1P_4$ – це $\frac{1}{2}[\overrightarrow{P_3P_1},\overrightarrow{P_3P_4}]$, аналогічно обчислюється площа $P_3P_2P_4$. Тобто $\frac{[\overrightarrow{P_3P_1},\overrightarrow{P_3P_4}]}{[\overrightarrow{P_3P_2},\overrightarrow{P_3P_4}]}=\frac{|P_1H_1|}{|P_2H_2|}=\frac{|P_1M|}{|P_2M|}$.



Мал. 9

Остатня рівність випливає з подібності трикутників P_1H_1M і P_2H_2M звідси отримуємо:

$$[\overrightarrow{P_3P_2},\overrightarrow{P_3P_4}]\cdot\overrightarrow{P_1M}=[\overrightarrow{P_3P_1},\overrightarrow{P_3P_4}]\cdot\overrightarrow{P_2M} \qquad \textbf{(12)}$$

В даній формулі враховано і те, що одна з площ може бути рівна нулю, і те, що вказані площі мають однакові знаки тоді, коли точки P_1 і P_2 лежать по одну сторону відносно прямої P_3P_4 (тобто коли вектор $\overrightarrow{P_1M}$ і $\overrightarrow{P_2M}$ співнапрямлені). Ми знайшли у якому відношенні ділиться даний відрізок точкою M. З іншої сторони, $\overrightarrow{P_2M} = \overrightarrow{P_1M} - \overrightarrow{P_1P_2}$. Підставимо ці співвідношення в (12). Виразимо $\overrightarrow{P_1M}$:

$$\overrightarrow{P_1M} = \frac{\left|\overrightarrow{P_3P_4}, \overrightarrow{P_3P_1}\right|}{\left|\overrightarrow{P_3P_2}, \overrightarrow{P_3P_4}\right| - \left|\overrightarrow{P_3P_1}, \overrightarrow{P_3P_4}\right|}\overrightarrow{P_1P_2} = \frac{\left|\overrightarrow{P_3P_4}, \overrightarrow{P_3P_1}\right|}{\left|\overrightarrow{P_3P_2} - \overrightarrow{P_3P_1}, \overrightarrow{P_3P_4}\right|}\overrightarrow{P_1P_2} = \frac{\left|\overrightarrow{P_3P_4}, \overrightarrow{P_3P_1}\right|}{\left|\overrightarrow{P_1P_2}, \overrightarrow{P_3P_4}\right|}\overrightarrow{P_1P_2} \qquad \textbf{(13)}$$

(Тут ми використали властивість лінійності косого добутку: $[\overrightarrow{P_3P_2}, \overrightarrow{P_3P_4}] - [\overrightarrow{P_3P_1}, \overrightarrow{P_3P_4}] = [\overrightarrow{P_3P_2} - \overrightarrow{P_3P_1}, \overrightarrow{P_3P_4}]$), яке легко вставити, якщо записати цю рівність в координатах). Тепер, знаючи координати точки P_1 і вектора $\overrightarrow{P_1M}$, знаходимо точку M.

Таким чином, ми знайшли точку перетину прямої і відрізка. Для знаходження перетину двох прямих на одній з них вибираються дві довільні точки P_1 і P_2, на другій – P_3 і P_4.

Можливо, виведення формули (13) не зовсім елементарне, але саму формулу легко запам'ятати, якщо зрозумілий її геометричний зміст. Так, якщо відрізки P_1P_2 і P_3P_4 перетинаються, то площа чотирикутника $P_1P_2P_3P_4$ за відомою формулою є половина добутку діагоналей на синус кута між ними. Тобто за абсолютною величиною косого добутку векторів $\overrightarrow{P_1P_2}$ і $\overrightarrow{P_3P_4}$, який стоїть в знаменнику формули (13), – це подвоєна площа чотирикутника $P_1P_2P_3P_4$. В чисельнику ми маємо подвоєну площу трикутника $P_3P_4P_1$.

2.4. ***Взаємне розміщення двох відрізків і променів.***

Перевірити перетин двох відрізків (як правило нас найчастіше цікавить лише сам факт перетину) не складно, коли знову ж використати косий добуток. Нехай перший відрізок заданий точками P_1 і P_2, а другий – P_3 і P_4. відрізки перетинаються тоді, коли, по-перше, перетинаються прямокутники, які обмежують дані відрізки, по-друге, кінці кожного відрізка лежать по різні сторони прямої на якій лежить інший відрізок. Перша умова дозволяє не розглядати окремий випадок, коли відрізки лежать на одній прямі. Позначимо x_{max1} і x_{min1} максимальну і мінімальну з x-координат першого відрізка, x_{max2} і x_{min2} – другого відрізка. Для y-координат аналогічно y_{max1}, y_{min1}, y_{max2}, y_{min2}. Тоді умову перетину формально можна записати:

1) $x_{max1} \geq x_{min2}$, $x_{max2} \geq x_{min1}$, $y_{max1} \geq y_{min2}$, $y_{max2} \geq y_{min1}$;
2) косі добутки $[\overrightarrow{P_1P_3}, \overrightarrow{P_1P_2}]$ і $[\overrightarrow{P_1P_4}, \overrightarrow{P_1P_2}]$ мають різні знаки, тобто $[\overrightarrow{P_1P_3}, \overrightarrow{P_1P_2}] \cdot [\overrightarrow{P_1P_4}, \overrightarrow{P_1P_2}] \leq 0$;
3) аналогічно $[\overrightarrow{P_3P_1}, \overrightarrow{P_3P_4}] \cdot [\overrightarrow{P_3P_2}, \overrightarrow{P_3P_4}] \leq 0$.

Якщо факт перетину двох відрізків встановлено, то точка перетину шукається, як і в задачі 2.3. Для відрізків однієї прямої їх перетин (точку або відрізок) можна знайти шляхом розрахунку значень двох скалярних добутків аналогічно задачі 2.1 або за допомогою порівняння координат кінців відрізка.

Введемо тепер поняття відстані між двома відрізками, які не перетинаються, як мінімальна серед відстаней між всіма парами точок двох відрізків. Нескладно зрозуміти, що відстань дорівнює відстані від кінця одного з відрізків до другого відрізка (задача 2.1). Тому для розв'язування даної задачі достатньо обчислити відповідні відстані для кожної з чотирьох точок кінців відрізків і вибрати з них мінімальне.

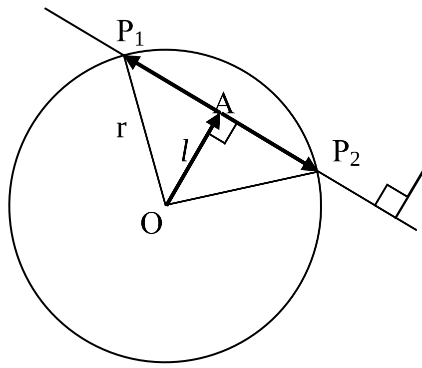
Для перевірки наявності перетину двох променів P_1P_2 і P_3P_4 потрібно розглянути взаємне розміщення двох прямих. Якщо косий добуток $[\overrightarrow{P_1P_2}, \overrightarrow{P_3P_4}]$ рівний нулю, це означає, що промені лежать на паралельних прямих. Якщо це дві різні прямі, то вектори $\overrightarrow{P_1P_3}$ і $\overrightarrow{P_1P_2}$ не колінеарні, значить, косий добуток $[\overrightarrow{P_1P_3}, \overrightarrow{P_1P_2}]$ не дорівнює нулю. В цьому випадку промені не перетинаються. Коли промені лежать на одній прямі, то за допомогою знаку скалярного добутку $(\overrightarrow{P_1P_2}, \overrightarrow{P_3P_4})$ можна зрозуміти, в одну чи різні сторони вони напрямлені. В першому випадку скалярний добуток буде додатнім, а в другому – від'ємний. Якщо промені співнапрямлені, то визначити, який із променів являється їх перетином, можна, обчислити значення скалярного добутку $(\overrightarrow{P_1P_2}, \overrightarrow{P_1P_3})$. Коли скалярний добуток більший нуля, перетином являється промінь P_3P_4, в протилежному випадку – промінь P_1P_2. У випадку, коли промені протилежнонапрямлені їх перетином може бути відрізок P_1P_3, і тоді початок кожного з променів лежить в середині другого: $(\overrightarrow{P_1P_2}, \overrightarrow{P_1P_3}) > 0$, або одна точка P_1=P_3: $(\overrightarrow{P_1P_2}, \overrightarrow{P_1P_3}) = 0$, або вони не мають спільних точок $(\overrightarrow{P_1P_2}, \overrightarrow{P_1P_3}) < 0$.

Якщо прямі P_1P_2 і P_3P_4 перетинаються в одній точці M ($[\overrightarrow{P_1P_3}, \overrightarrow{P_1P_2}] \neq 0$), то знайти цю точку можна, як і в задачі 2.3. Пізніше потрібно перевірити, чи т. M належить кожному із променів: $(\overrightarrow{P_1P_2}, \overrightarrow{P_1M}) \geq 0$ і $(\overrightarrow{P_3P_4}, \overrightarrow{P_3M}) \geq 0$.

2.5. ***Взаємне розміщення кола і прямої.***

Пряма може перетинати коло в двох точках, дотикатись до нього, або не мати спільних точок з колом. Дані випадки легко визначити. Потрібно знайти відстань від центру кола до даної прямої (за допомогою формули відстані від точки до прямої, див. 2.1). Якщо дана відстань (позначимо її *l*) менша за радіус кола r, то пряма перетинає коло в двох точках, якщо рівна йому, то пряма дотикається до кола, а якщо більша за радіус, то спільних точок вони не мають. В останньому випадку нас може цікавити відстань від прямої до кола. Вона рівна *l*- r.

Більш складнішою є задача пошуку спільних точок прямої і кола. Випадок дотичної був розглянутий нами в п. 1.7. Нехай тепер пряма перетинає коло в двох точках. Координати цих точок можна знайти за наступним алгоритмом (мал. 10). Знайдемо вектор ***n*** нормалі до прямої. Відкладемо в напрямку цього вектора вектор $\overrightarrow{OA}$ довжини *l*. Обчислимо відстань $|AP_1| = |AP_2| = \sqrt{r^2 - l^2}$. Від т. A вздовж прямої відкладемо в обидва боки вектори довжини $|AP_1|$. Їх кінці дадуть нам дві шукані точки P_1 і P_2. Кожний крок даного алгоритму розглядався нами окремо раніше. Зауважимо тільки те, що на першому кроці необхідно правильно вибрати один з двох можливих напрямків нормалі до прямої. Для цього достатньо перевірити, що скалярний добуток $(n,\overrightarrow{OM}) \geq 0$, де M – довільна точка прямої.



Мал. 10

2.6. ***Взаємне розміщення двох кіл.***

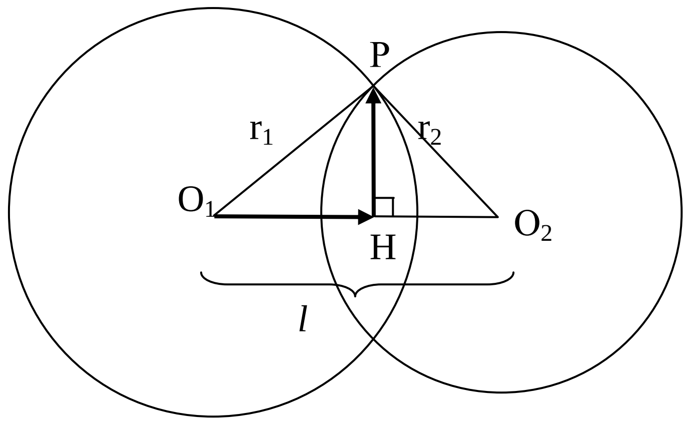
Два різних кола також можуть перетинатися в двох точках, дотикатися одне до одного і не мати спільних точок. В останньому випадку одне із кіл може лежати в середині іншого (назвемо такі кола вкладеними) або кожне із кіл лежить зовні іншого.

Перевірка перетину або дотику аналогічна попередній задачі здійснюється шляхом порівняння відстаней між центрами кіл (позначимо її *l*) з їх радіусами, якщо $l > r_1 + r_2$ або $l < |r_1 - r_2|$, то кола спільних точок не мають. Друга умова показує, що одне коло лежить всередині іншого. При заміні знаку в будь-якій з цих двох нерівностей на рівність, ми отримаємо дотикання даних кіл (зовнішнє або внутрішнє дотикання). Якщо $|r_1 - r_2| < l < r_1 + r_2$, то кола мають дві точки перетину.

Координати точки дотику кіл знайти дуже просто. Центри кіл $O_1(x_1, y_1)$ і $O_2(x_2, y_2)$ задають пряму на якій лежить і точка дотику P (x_3, y_3). Будемо вважати, що т. O_1 є центром кола з більшим радіусом. Тоді вектор $\overrightarrow{O_1P}$ співнапрямлений з вектором $\overrightarrow{O_1O_2}$. Довжини обох векторів також відомі, тому шукані координати дорівнюють: $\left(\frac{x_1 + (x_2 - x_1)r_1}{l}, \frac{y_1 + (y_2 - y_1)r_1}{l}\right)$.

Перевірте, якщо $r_1 < r_2$ то у випадку вкладення кіл отримана формула потребує корекції.

Для пошуку координат двох точок перетину кіл використаємо механізм, вже описаний в задачі 1.7. Розглянемо трикутник O_1O_2P (мал. 11). В трикутнику відомі всі три довжини сторін ($r_1 > r_2$ і l). Проведемо в трикутнику висоту PH. В отриманому прямокутному трикутнику O_1HP невідомі довжини катетів. Знайдемо O_1H, записавши теорему косинусів для трикутника O_1O_2P: $r_2^2 = r_1^2 + l^2 - 2lr_1 \cdot \cos\varphi = r_1^2 + l^2 - 2l \cdot |O_1H|$. Звідси, $|O_1H| = \dfrac{(r_1^2 + l^2 - r_2^2)}{2l}$. За теоремою Піфагора $|HP| = \sqrt{r_1^2 - |O_1H|^2}$. Тепер послідовно знаходимо: вектор $\overrightarrow{O_1H} = \dfrac{|O_1H|}{l}\overrightarrow{O_1O_2}$, точку H за відомою т. O_1 і вектором $\overrightarrow{O_1H}$, вектор $\boldsymbol{n}=(y_2-y_1, x_1-x_2)$, перпендикулярний O_1O_2, вектор $\overrightarrow{HP} = \dfrac{|HP|}{|n|}n$, а також точку P за відомою точкою H і вектором $\overrightarrow{HP}$. Замінивши в остатній дії вектор $\overrightarrow{HP}$ на протилежний, отримаємо і другу точку перетину кіл.



Мал. 11

Зауважимо, що при розв'язувані задач 1.7, 2.5 і 2.6 ми використовуємо «покрокові» алгоритми. А саме, аналізуючи малюнок, виділяємо ланцюжок об'єктів (вектори точки та ін.). Фактично ми послідовно встановлюємо параметри об'єктів виділених нами на малюнку. Ланцюжок обчислень повинен привести нас до відповіді, при цьому кожен крок дуже простий і повністю очевидний. Не виключено, що задача має і більш коротший аналітичний розв'язок (наприклад, розв'язування системи рівнянь). Але «покроковий» метод в силу своєї наочності, дозволяє контролювати кожний крок розв'язку, а саме ця властивість може відіграти важливу роль при перевірці алгоритму і налагодженні програми.

2.7. *Перевірка належності точки внутрішній області багатокутника.*

Нехай M – деяка точка площини. Потрібно визначити її місцезнаходження відносно замкнутої ламаної. Розглянемо спочатку випадок опуклого багатокутника. Нехай задані координати вершин багатокутника $P_0, P_1, \ldots, P_{n-1}$ перераховані в порядку його обходу проти годинникової стрілки. При такому обході багатокутник лежить зліва від границі. Отже, якщо точка M лежить всередині багатокутника, то орієнтований кут між векторами $\overrightarrow{P_i M}$ і $\overrightarrow{P_i P_{i+1}}$ від'ємний. Тому нам достатньо обчислити величини косих добутків $[\overrightarrow{P_i M}, \overrightarrow{P_i P_{i+1}}]$, i=0,1,…,n-1; значення i+1 береться по модулю n. Якщо всі отримані при цьому значення менші нуля, то точка M внутрішня. Якщо ж одне з них рівне нулю, а всі інші менші нуля, тоді M належить границі багатокутника (переконайтесь, що просто рівність нулю одного із значень недостатньо). В протилежному випадку точка M лежить поза межами багатокутника.

Розглянемо тепер довільний багатокутник. Проведемо горизонтальний промінь з точки M, наприклад, вліво. Так, як багатокутник обмежений, то завжди легко взяти на промені точку P(x,y), яка не належить йому. Далі підрахуємо кількість перетинів відрізка PM з границею багатокутника. Якщо ця кількість рівна нулю, або парна, то точка M лежить за межами багатокутника, в протилежному випадку – всередині нього.

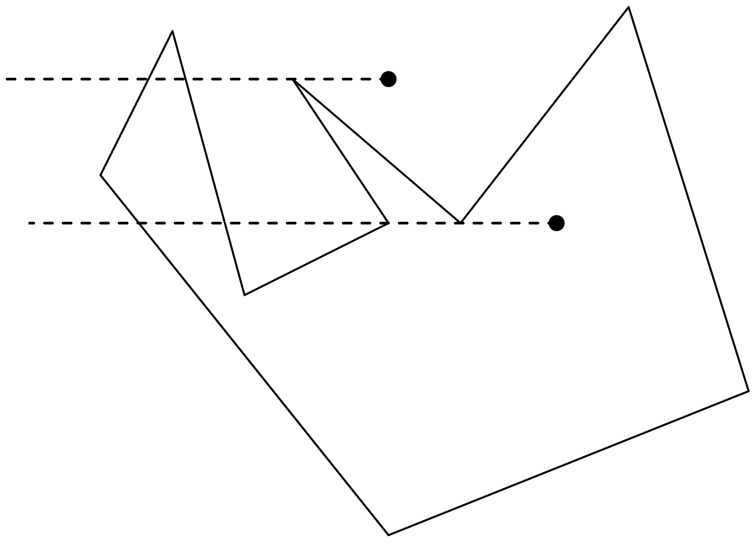
Кількість перетинів відрізка PM з границею ми підрахуємо, розглянувши по черзі перетин відрізка PM з кожною ланкою ламаної. При цьому можливі такі випадки.

1) Одна з ланок ламаної повністю лежить на відрізку PM.

2) Ланка ламаної дотикається до відрізка PM.

3) M лежить на одній із ланок ламаної.

В остатньому випадку M належить границі багатокутника, і підрахунку загального числа перетинів не потребує. Для двох перших випадків зробимо наступне: в першому випадку на перетин не будемо звертати увагу. В другому – додатково перевіримо «нижнім» чи «верхнім» кінцем ланки ламаної дотикається до відрізка PM. Якщо точка дотику є «нижнім» кінцем ланки, то перетин ігнорується, а якщо «верхнім» - то зараховується. Враховуючи це, отримаємо, що дотик відрізка PM до границі багатокутника в одних точках ігнорується, а в інших точках рахується двічі. Це не змінить парності числа перетинів, а тільки вона важлива при пошуку відповіді в даній задачі. Якщо ж відрізок дійсно перетинає ламану в її вершині, то, за нашою домовленістю, число перетинів якраз збільшиться на одиницю (перетин з верхнім ребром зарахований не буде, а з нижнім – буде). Наприклад, на мал.12 кількість перетинів дорівнює чотирьом (дотик зараховано два рази), а для нижньої точки – трьом (дотик неврахований, а перетин у вершині ламаної врахований лише один раз).



Мал. 12

ОСОБЛИВІ ТОЧКИ БАГАТОКУТНИКІВ
І МНОЖИН N ТОЧОК ПЛОЩИНИ

3.1. ***Побудова кола, описаного навколо трикутника чи правильного n-кутника.***

Нехай трикутник або правильний n-кутник заданий координатами своїх вершин. Для розв’язання задачі нам достатньо знайти координати центру кола, тоді його радіус R виразимо через координати центру і координати будь-якої з його вершин. Зауважимо, що задача знаходження радіусу такого кола є простою для трикутника $P_1P_2P_3$ і спирається на співставлення двох формул для площі трикутника: $S=\frac{\left|\left[\overrightarrow{P_1P_2},\overrightarrow{P_1P_3}\right]\right|}{2}=\frac{|P_1P_2|\cdot|P_1P_3|\cdot|P_2P_3|}{4R}$, нагадаємо, що перший спосіб випливає з геометричного змісту косого добутку.

З курсу геометрії відомо, що центр описаного кола лежить на перетині серединних перпендикулярів проведених до сторін трикутника. Координати середини сторони трикутника є середнім арифметичним координат відповідних векторів. Тоді задача знаходження рівняння серединного перпендикуляра співпадає з задачею 1.3 і за формулою (7) ми маємо: $(x_2-x_1)\left(x-\frac{x_1+x_2}{2}\right)+(y_2-y_1)\left(y-\frac{y_1+y_2}{2}\right)=0$, де (x_i,y_i) – координати точки P_i. Для знаходження центру шуканого кола достатньо записати ще один серединний перпендикуляр (наприклад, до відрізка P_1P_3) і знайти точку перетину цих двох прямих.

Для правильного багатокутника розв’язування нічим не відрізняється від наведеного вище. Достатньо записати рівняння двох будь-яких не співпадаючих серединних перпендикулярів, що не співпадають. Бо саме перетин всіх серединних перпендикулярів до сторін багатокутника в одній точці дозволяє описати коло навколо правильного n-кутника. Але є кращий спосіб. У випадку, якщо N – парне, центр правильного N-кутника – це середина проведеної діагоналі, наприклад з першої вершини в $\left(\frac{N}{2}+1\right)-y$, тобто координати центра кола є середнім арифметичним координат зазначених вершин. Якщо N – непарне, то центр лежить на прямій, яка з'єднує одну із вершин з серединою найбільш віддаленої сторони, на відстані $R = \dfrac{a}{2\sin\left(\dfrac{\pi}{n}\right)}$ від вершини, де а – довжина сторони правильного n-кутника.

Зауважимо, що в правильному багатокутнику центр описаного кола співпадає з центром одиничних мас, розміщених в його вершинах. Тому його розташування також можна обчислити за формулою (15) див. 3.7 (далі).

3.2. ***Побудова кола вписаного в трикутник або правильний n-кутник.***

Як і в попередній задачі, радіус r такого кола для трикутника легко знайти із співвідношення формул обчислення площі трикутника: $S = \dfrac{\left|\left[\overrightarrow{P_1P_2}, \overrightarrow{P_1P_3}\right]\right|}{2} = r\dfrac{|P_1P_2| + |P_1P_3| + |P_2P_3|}{2}$.

Центр вписаного в трикутник кола лежить на перетині бісектрис його кутів. Для знаходження його координат достатньо записати рівняння будь-яких двох бісектрис (див. задача 1.5) і знайти точку їх перетину.

Центр кола, вписаного в правильний багатокутник, співпадає з центром описаного кола, а його радіус рівний відстані від центру до будь-якої із сторін. Як і з описаним колом, шуканий радіус можна обчислити відразу, не шукаючи центру, за формулою: $r = \dfrac{a}{2tg\left(\dfrac{\pi}{N}\right)}$.

3.3. ***Коло, що «охоплює» N точок площини.***

Ця задача полягає у відшуканні координат центру кола мінімально можливого радіусу, всередині якого знаходяться всі задані точки. Інколи дану задачу називають мінімаксимальною задачею «про культурний центр». В цій задачі потрібно за координатами будівель міста підібрати місце для будівництва культурного центру так, щоб відстань до максимально віддаленого від нього будинку була мінімальною. Для того щоб зрозуміти розв’язок даної задачі в загальному випадку, розглянемо спочатку «трикутний» варіант: N=3.

Навіть для трьох точок розв’язок залежить від їх взаємного розміщення. Нехай точки лежать на одній прямій або утворюють тупокутний трикутник. Тоді шукана точка лежить на середині відрізка, який з’єднує найбільш віддалені точки (на середині найбільшої сторони трикутника). Справді, відстань від цієї точки до будь-якої з перших двох зменшити неможливо, а третя точка знаходиться на меншій відстані від знайденої точки, тому вона лежить всередині кола, діаметр, якого утворює дві інші точки. А для гострокутного трикутника розв’язком буде центр описаного кола (зміщення шуканої точки в будь-якому напрямку призведе до збільшення відстані хоча б до однієї з точок). Спосіб його знаходження був показаний в задачі 3.1. Прямокутний трикутник є «граничним» для цих двох випадків, тобто для нього шукану точку можна шукати будь-яким із описаних способів (але перший спосіб є простішим).

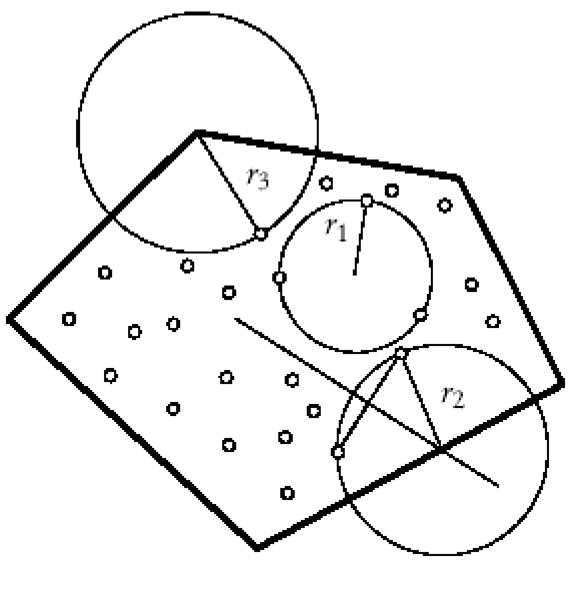
Для довільного N також є два випадки. Якщо знайдеться такі дві точки, що коло, побудоване на відрізку, який з’єднує ці точки, як на діаметрі, містить в собі всі інші точки (тобто для них виконується нерівність $(x_i - x_0)^2 + (y_i - y_0)^2 \le r^2$, де (x_0, y_0) – центр кола), то це коло – шукане (фактично це є випадок «тупокутного трикутника»). Якщо ж такої пари точок не знайдеться, то шукане коло буде проходити хоча б через три точки. Тому тепер потрібно перебрати всі трійки точок до тих пір, поки не знайдеться така трійка, що проходячи через них, коло охопить всередині себе всі інші точки (випадок «гострокутного трикутника»).

3.4. ***Найбільше «порожнє» коло з центром всередині багатокутника, який містить N точок площини.***

Дану задачу можна сформулювати, як задачу «про хімічний завод». А саме, в межах міста, границя якого відома, а будинки задані своїми координатами, потрібно вибрати місце для будівництва хімічного заводу так, щоб відстань від нього до найближчого будинку була максимальною.

Фактично нам потрібно знайти коло з максимальним радіусом, яке не має всередині точок з початкової множини, центр якого лежить всередині або на границі заданої ламаної.

Тут можливі три випадки розміщення центру шуканого кола. Спочатку припустимо, що він лежить всередині ламаної. Тоді коло проходить обов'язково через три точки заданої множини, інакше, знайдеться «порожнє» коло і більшого радіусу. Тому переберемо всі не колінеарні трійки точок і для кожної трійки розглянемо коло, яке через них проходить. З цих кіл виберемо ті, центр яких лежить всередині ламаної і не містить в собі інших точок. Знайдемо серед них коло з найбільшим радіусом (позначимо його r_1). У другому випадку центр шуканого кола лежить на ламаній, але не співпадає з її вершинами. Шукане коло тут визначається вже парою точок. Центри таких кіл лежать на перетині ламаної із серединним перпендикуляром до відрізка, який з'єднує дві точки (мал. 13). Позначимо максимальний радіус «порожніх» кіл даного виду r_2. Тепер, якщо центр шуканого кола співпадає з однією із вершин ламаної, то його радіус буде визначатись відстанню найближчої до вершини точки (максимальний з даних радіусів позначимо r_3). Відповіддю до нашої задачі буде радіус одного з трьох знайдених кіл, який є $\max(r_1,r_2,r_3)$. Пошук розв'язку можна здійснити за O(NlogN) операцій.



Мал. 13

Наш алгоритм має допустиму обчислювальну складність, але при його реалізації доводиться використовувати відразу декілька елементарних задач, розглянутих вище. Як правило, для школярів повне розв'язання є дуже складним. Тому цього покажемо наближений метод розв'язку даної задачі, трохи спростивши її. Припустимо, що ламана являє собою прямокутник, лівий нижній кут якого розташований в початку координат, координати правого верхнього (x_r,y_r) відповідають довжині і ширині прямокутника. Це спрощення не зменшує загальності, оскільки навколо кожного багатокутника можна описати прямокутник і розв'язати дану задачу для цього прямокутника. Центри кіл, які лежать за межами багатокутника, розглядати не потрібно.

Нехай ми хочемо знайти «порожнє» коло радіуса r. Тоді, щоб перевірити, чи ми зможемо це зробити, виконаємо наступну операцію: побудуємо кола радіуса r з центром в кожній з точок. Якщо ці кола покривають прямокутник повністю, то очевидно, що «порожнього» кола вказаного радіуса не існує. Будь-яка не покрита такими колами точка прямокутника може бути центром «порожнього» кола. Даний метод в геометрії називається методом «роздуття». Ми звели початкову задачу до іншої: задачі про покриття.

3.5. ***Задача про покриття.***

Припустимо, ми хочемо перевірити, що деякий прямокутник повністю покривається заданою множиною кіл. Якщо всі чотири його вершини покриваються одним колом, то прямокутник покривається колами повністю. В противному випадку розіб'ємо прямокутник на чотири однакових прямокутники і рекурсивно перевіримо, що кожний з них покривається колами. Для цього прямокутники, які не містяться повністю всередині якого-небудь одного кола, знову будемо ділити на чотири частини. Виключенням буде випадок, при якому вершина розглянутого прямокутника буде поза межами всіх кіл, тобто буде прикладом не покритої точки. Будемо продовжувати розбиття, поки сторона прямокутника не стане меншою за деяку задану достатньо малу величину. Тоді припустимо, що даний прямокутник повністю колами не покривається колом, а його центр будемо вважати не покритою точкою.

Рекурсивна функція *Check*, що виконує відповідну перевірку, наведена нижче:

```
const
eps1=1e-6; {точність пошуку непокритої точки}
eps2=1e-5; {точність пошуку радіуса}
var fx, fy: real;
{координати цетру кола}
xr, yr: real; {розміри прямокутника}
lb, rb,r: real; {r – шуканий радіус}
function dist2(x1, y1, x2, y2: real): real;
{обчислює квадрат відстані між двома точками}
begin
dist2:=sqr(x1-x2)+sqr(y1-y2)
end;
function check(x1, y1, x2, y2: real): boolean;
{перевіряє, чи прямокутник покривається заданою множиною кругів;
параметри – координати лівого верхнього і правого нижнього кутів
прямокутника}
var i: longint;
d1, d2, d3, d4, c1, c2, c3, c4: boolean;
begin
if (abs(x1-x2)<eps1 and
(abs(y1-y2)<eps1 then
begin {центр прямокутника – непокрита точка}
fx:=(x1+x2)/2;
fy:=(y1+y2)/2;
check:=false; exit
end;
check:=true;
c1:= true; c2:= true;
c3:= true; c4:= true;
{перевіряємо покриття одним кругом}
for i:=1 to n do
begin
d1:=dist2(x1, y1,x[i], y[i])<=r*r);
d2:=dist2(x1, y2,x[i], y[i])<=r*r);
d3:=dist2(x2, y1,x[i], y[i])<=r*r);
d4:=dist2(x2, y2,x[i], y[i])<=r*r);
if d1 and d2 and d3 and d4 then
```

```
        begin
        check:=true; exit
        end;
        c1:=c1 and not d1;
        c2:=c2 and not d2;
        c3:=c3 and not d3;
        c4:=c4 and not d4
        end;
        if c1 then {точка x1, y1 не покрита}
        begin
        fx:=x1; fy:=y1;
        check:=false; exit
        end;
        if c2 then {точка x1, y2 не покрита}
        begin
        fx:=x1; fy:=y2;
        check:=false; exit
        end;
        if c3 then {точка x2, y1 не покрита}
        begin
        fx:=x2; fy:=y1;
        check:=false; exit
        end;
        if c4 then {точка x2, y2 не покрита}
        begin
        fx:=x2; fy:=y2;
        check:=false; exit
        end;
        check:=check(x1, y1, (x1+x2)/2, (y1+y2)/2) and
                  check((x1+x2)/2, y1, x2, (y1+y2)/2) and
                  check(x1, (y1+y2)/2, (x1+x2)/2,  y2) and
                  check((x1+x2)/2, (y1+y2)/2, x2, y2)
end;
```

Тепер ми легко можемо розв’язати і задачу про «порожнє» коло максимального радіуса. За допомогою функції *Check* шуканий радіус кола також можна знайти алгоритмом ділення пополам:

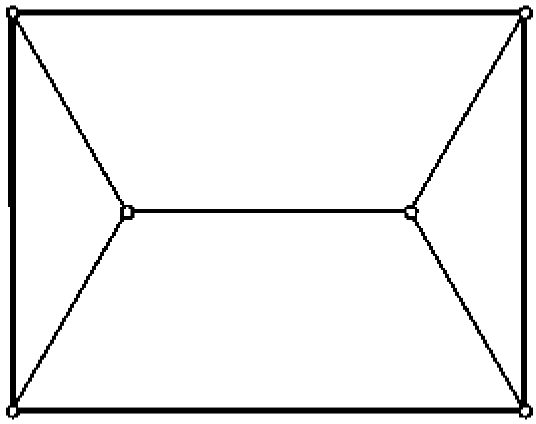
```
lb:=0; {ліва межа}
if xr<yr then rb:=xr/2
        else rb:=yr/2; {права межа}
while abs(lb-rb)>eps2 do
    begin
    r:=(lb+rb)/2;
    if check(r, 0,0, xr, yr) then rb:=m
```

```
else lb:=m
end;
writeln (fx:0:4, '', fy:0:4, '', r:0:4);
```

Таким не складним способом можна розв'язати задачу з будь-якою точністю.

3.6. ***Найкоротша мережа доріг.***

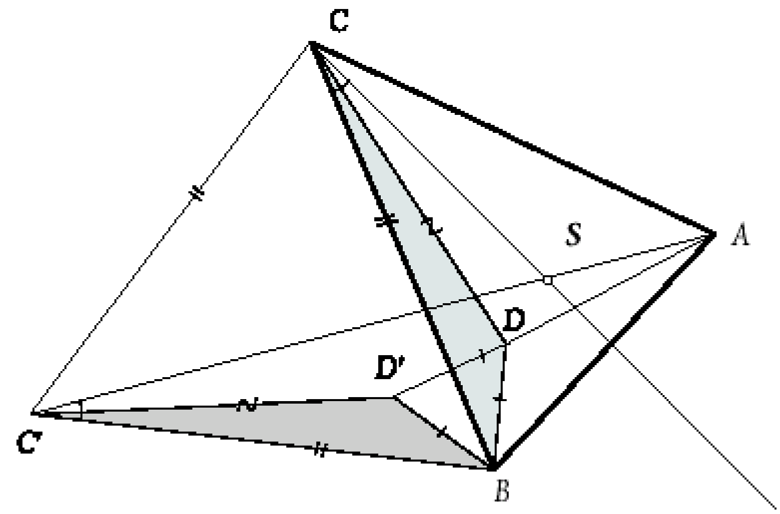
Задані N населених пунктів (точок на площині). Необхідно прокласти між ними дороги так, щоб по цих дорогам можна проїхати з будь-якого пункту в будь-який інший, а сумарна довжина доріг була мінімальною. На відміну від подібної задачі побудови мінімального остова в теорії графів в даній задачі ми не обмежені відрізками прямих, які з'єднують задані точки. При необхідності ми можемо побудувати в довільних місцях площини нові точки перетину частин доріг (так, для чотирьох точок, розташованих у вершинах квадрата, система доріг, побудована з двох діагоналей цього квадрату, краща за любе основне дерево, але і вона не являється оптимальною, див. на мал. 14 розв'язання для прямокутника.



Мал. 14

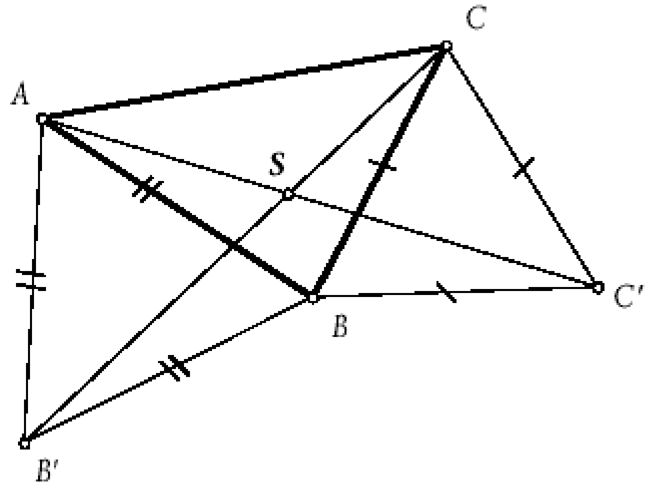
Розглянемо розв'язання задачі для N=3 (населені пункти лежать у вершинах трикутника ABC).

Неважко зрозуміти, що в цьому випадку задача зводиться до пошуку точки, сума відстаней від якої до всіх вершин трикутника мінімальна, і така точка повинна лежати всередині або на стороні трикутника ABC. Припустимо, що кожний з кутів трикутника ABC не більший за 120^o. Нехай D – довільна точка. Розглянемо трикутник $BC'D'$, отриманий поворотом трикутника BCD навколо т. B на 60° (мал. 15). За побудовою $DC = D'C'$ і $DD' = BD$ (трикутник BDD' – рівносторонній). Тому шукана сума відстаней для т. D дорівнює $AD + BD + CD = AD + DD' + D'C'$ – і, отже, нам потрібно знайти таку т. D, для якої довжина ламаної $ADD'C'$ мінімальна. Якщо в якості т. D ми виберемо т. S, див. на мал. 15 ($\angle AC'B = \angle SCB$), то після повороту навколо точки B на 60° вона попадає на відрізок $C'S$. Таким чином, довжина $ASS'C'$ буде рівною AC', і зрозуміло, не більша довжини любої другої ламаної $ADD'C'$. Звідси S і є шукана точка. Вона називається «точкою Штейнера». Відмітимо, що $\angle CSA = 120^{o}$, так як промінь CS переходить в промінь $C'A$ при повороті на 60°. Зрозуміло, що дві інші сторони трикутника повинно бути видно з точки S під кутом 120°.



Мал. 15

З проведеного аналізу слідує і спосіб побудови точки Штейнера (мал. 16). Спочатку шукаємо точки B' і C' як вершини рівносторонніх трикутників ABB' і BCC', побудованих за межами трикутника ABC. Пошук їх координат здійснюється за допомогою вектора нормалі відкладеного до середини відповідної сторони трикутника (шукана точка знаходиться на відстані $\frac{\sqrt{3}}{2}a$ від середини сторони початкового трикутника, де а – довжина відповідної сторони). Залишається визначити т. S перетину відрізків CB' і AC' (див 2.3).



Мал. 16

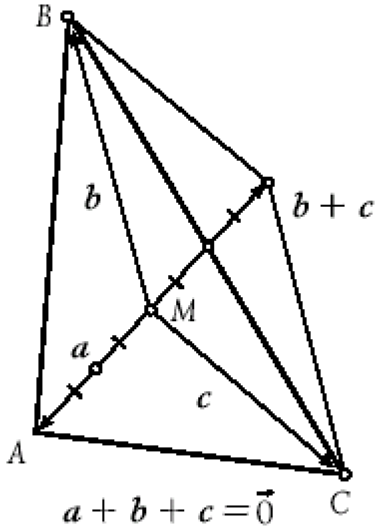
Для трикутників, у яких один з кутів більше 120^o (в тому числі виродженого у відрізок), запропонована нами побудова не підходить. Дійсно, в такому трикутнику не має точки з якої всі три сторони було б видно під кутом 120^o. В даному випадку розв'язком задачі буде система з двох найменших сторін трикутника. На мал. 14 показано розв'язок нашої задачі для чотирьох точок, розміщених у вершинах прямокутника. Для довільних N точок задача про найкоротшу мережу доріг не розв'язана. Тому пошук мінімальної транспортної мережі здійснюється з використанням комп'ютера. Але всі відомі на сьогодні алгоритми дозволяють побудувати розв'язок лише при невеликих значеннях N.

3.7. ***Центр мас***.

В деяких задачах геометрії корисно використовувати центр мас. Нехай на площині є система матеріальних точок $A_1,A_2,\ldots,A_N$, які мають відповідні маси $m_1,m_2,\ldots,m_N$. Будемо рахувати, що маси – не від'ємні числа (деколи допускаються від'ємні і навіть комплексні маси). Центром мас такої системи матеріальних точок називається точка M, для якої

$$m_1\overrightarrow{MA_1}+m_2\overrightarrow{MA_2}+\ldots+m_N\overrightarrow{MA_N}=\vec{0} \quad \textbf{(14)}$$

Простий випадок: точка перетину медіан в трикутнику є центром трьох рівних мас, розміщених у вершинах даного трикутника. Це випливає з теореми про медіани (мал. 17)



Мал. 17

Повернемся до N точок. Нехай O – довільна точка площини. Використовуючи визначення точки M, отримаємо:

$$\begin{aligned}
&m_1\overrightarrow{OA_1}+m_2\overrightarrow{OA_2}+\ldots+m_N\overrightarrow{OA_N}=\\
&=m_1(\overrightarrow{OM}+\overrightarrow{MA_1})+m_2(\overrightarrow{OM}+\overrightarrow{MA_2})+\ldots+m_N(\overrightarrow{OM}+\overrightarrow{OA_N})=\\
&=(m_1+m_2+\ldots+m_N)\overrightarrow{OM}+m_1\overrightarrow{MA_1}+m_2\overrightarrow{MA_2}+\ldots+m_N\overrightarrow{MA_N}=\\
&=(m_1+m_2+\ldots+m_N)\overrightarrow{OM}
\end{aligned}$$

В якості точки O можна взяти початок координат. Тоді знайдемо координати точки M:

$$\begin{aligned}
x_M&=\frac{m_1x_1+m_2x_2+\ldots+m_Nx_N}{m_1+m_2+\ldots+m_N},\\
y_M&=\frac{m_1y_1+m_2y_2+\ldots+m_Ny_N}{m_1+m_2+\ldots+m_N}.
\end{aligned} \qquad \textbf{(15)}$$

– де (x_i, y_i) – координати A_i.

Застосування центру мас основується на його означенні (14). Покажемо це на одному з прикладів.

Задача «мережа»

Губернатор однієї з областей заключив договір з фірмою «GopNet» контракт на підключення всіх міст області до комп’ютерної мережі. Мережа створюється наступним чином: в області встановлюється станція супутникового зв’язку, а після чого з кожного міста прокладається кабель до станції. Технологія, яка використовується компанією, для виключення накопичення помилок вимагає при збільшенні відстані збільшення товщини кабелю. Ціна кабелю, який поєднує міста з станцією, при використанні компанією даної технології дорівнює kL^2, де L – відстань від міста до станції, а k - деякий коефіцієнт. Потрібно визначити розміщення станції, при якому затрати компанії на побудову мережі будуть мінімальні.

Розмістимо в кожне місто масу k, і нехай M – центр мас отриманої системи рівних точкових мас. Якщо станцію встановити в точці P, то вартість S всього кабелю обчислюється таким чином:

$$S = \sum k\left|\overrightarrow{PA_i}\right|^2 = k\sum(\overrightarrow{PM} + \overrightarrow{MA_i}, \overrightarrow{PM} + \overrightarrow{MA_i}) =$$
$$= k\sum\left|\overrightarrow{PM}\right|^2 + k\sum(\overrightarrow{PM}, \overrightarrow{MA_i}) + k\sum\left|\overrightarrow{MA_i}\right|^2 =$$
$$= k\sum\left|\overrightarrow{PM}\right|^2 + (\overrightarrow{PM}, k\sum\overrightarrow{MA_i}) + k\sum\left|\overrightarrow{MA_i}\right|^2 =$$
$$= k\sum\left|\overrightarrow{PM}\right|^2 + k\sum\left|\overrightarrow{PA_i}\right|^2.$$

Звичайно величина S мінімальна, коли $\overrightarrow{PM} = \vec{0}$, тобто, коли станція знаходиться в центрі мас. Отже, координати шуканої точки шукаємо за формулою (15): $\left(\frac{x_1 + x_2 + ... + x_N}{N}, \frac{y_1 + y_2 + ... + y_N}{N}\right)$. Крім того, можна уявити, що по якійсь причині коефіцієнт k залежить від місцевості і є різним для кожного міста. Наш розв'язок легко адаптувати до цього випадку: треба розмістити в кожне місто масу, яка рівна відповідному коефіцієнту. Але знову ж станцію потрібно ставити в центрі мас отриманої системи матеріальних точок.

БАГАТОКУТНИКИ

4.1. ***Визначення виду трикутника.***

Спочатку розглянемо, як для трьох дійсних чисел a, b і c перевірити, чи існує трикутник з довжинами сторін, які відповідають заданим числам. Відомо для додатних a, b і c повинні виконуватись умови існування трикутника a+b>c, a+c>b і b+c>a. Перевірки цих нерівностей достатньо у будь-якому випадку. Дійсно, додавши перші дві нерівності, отримаємо

2a+b+c>b+c, звідси a>0. З інших пар нерівностей випливає b>0 і c>0. Звідси, перевірку знаку a, b і c виконувати не потрібно.

Якщо трикутник заданий координатами своїх вершин, то обчислювати довжини сторін і перевіряти існування трикутника не потрібно. В цьому випадку трикутник не існує тільки тоді, коли дані точки лежать на одній прямій. Це можна перевірити, обчисливши значення косого добутку $\left[\overrightarrow{P_1P_2}, \overrightarrow{P_2P_3}\right]$. Якщо він рівний нулю, то відповідні вектори колінеарні, тобто всі три точки лежать на одній прямій.

Як виянити за координатами вершин трикутника, чи гострокутний він, чи прямокутний або чи тупокутний? Для цього достатньо знати, яким є найбільший його кут. Вид кута легко визначити за знаком скалярного добутку утворюючих його векторів: більше нуля для гострого кута, рівне нулю для прямого і менше нуля для тупого. Тому підрахуємо всі три скалярних добутки, перемножимо їх і за знаком добутку визначимо вид трикутника. Зверніть увагу, що значення самих кутів обчислювати не потрібно.

4.2. ***Перевірка опуклості багатокутника.***

Опуклість багатокутника з вершинами $P_1, P_2, \ldots, P_n$, перерахованих в порядку його обходу, легко перевірити, якщо визначити знаки косих добутків $\left[\overrightarrow{P_iP_{i+1}}, \overrightarrow{P_{i+1}P_{i+2}}\right]$, i=1,..,n (тут P_{n+1} є P_1, а P_{n+2} – P_2). В опуклого багатокутника знаки всіх добутків або не додатні, або не від’ємні (тобто знаки ненульових добутків співпадають). Якщо ми знаємо напрямок обходу, то знак косих добутків для опуклого багатокутника визначений: при обході за годинниковою стрілкою всі косі добутки не додатні, а проти годинникової стрілки не від’ємні.

4.3. ***Обчислення площі простого багатокутника.***

Під простим багатокутником ми розуміємо такий багатокутник, границя якого не має самоперетинів і самодотикань. Нехай вершини $P_1, P_2, \ldots, P_n$ простого багатокутника перераховані за порядком обходу його границі. Нагадаємо, що для обчислення його орієнтованої площі потрібно додати орієнтовані площі трикутників OP_1P_2, OP_2P_3,…, OP_nP_1, де O – довільна точка площини. Модуль отриманої величини і є шукана площа:

$$S = \frac{1}{2}\left|\sum_{i=1}^{n}[\overrightarrow{OP_i}, \overrightarrow{OP_{i+1}}]\right| \qquad \textbf{(16)}$$

– де вважається $P_{n+1}=P_1$. В якості точки O в деяких задачах буває доцільно вибрати одну з вершин багатокутника. Якщо ж за точку O взяти початок координат, то формула (16) набуде іншого вигляду, також зручного для програмування:

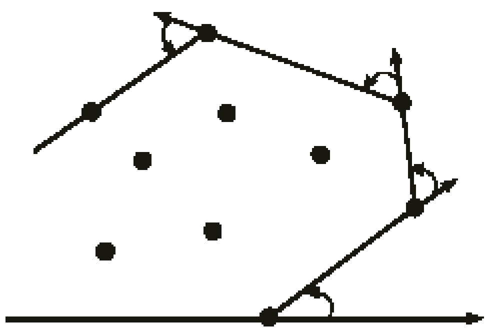
$$S = \frac{1}{2}\left|(x_1y_2 - x_2y_1) + (x_2y_3 - x_3y_2) + \ldots + (x_ny_1 - x_1y_n)\right| =$$
$$\frac{1}{2}\left|x_1(y_2 - y_n) + x_2(y_3 - y_1) + x_3(y_4 - y_2) + \ldots + x_n(y_1 - y_{n-1})\right| \qquad \textbf{(17)}$$

– де (x_i, y_i) – координати точки P_i.

4.4. ***Побудова опуклої оболонки для множини з N точок площини.***

Опуклою оболонкою деякої заданої множини точок називається перетин всіх опуклих множин, що містять задану множину. Для скінченої множини точок опуклою оболонкою буде завжди опуклий багатокутник, всі вершини якого є точками заданої множини.

Задача полягає в тому, щоб для заданої скінченої множини точок знайти вершини опуклої оболонки цієї множини. Будемо перераховувати вершини в порядку перегляду проти годинникової стрілки. Для ефективного розв’язування цієї задачі існує декілька різних алгоритмів. Наведемо найбільш просту реалізацію одного з них – алгоритму Джарвіса. Цей алгоритм інколи називають «загортання подарунка» (див. мал. 18).



Мал. 18

Перерахунок точок шуканої межі опуклого багатокутника почнемо з правої нижньої точки P_1, яка належить межі опуклої оболонки. Позначимо її координати (x_1, y_1). Наступною при заданому обході буде точка $P_2(x_2, y_2)$. Вона має таку властивість, яку мають всі інші точки , що лежать зліва від вектора $\overrightarrow{P_1P_2}$, тобто орієнтований кут між векторами $\overrightarrow{P_1P_2}$ і $\overrightarrow{P_1M}$ невід'ємний для будь-якої іншої точки M нашої множини. Для претендента на роль точки P_2 перевіримо виконання умови $[\overrightarrow{P_1P_2}, \overrightarrow{P_1M}]$? 0 зі всіма точками M. Якщо точок, що задовольняють цю умову, декілька, то вершиною шуканого багатокутника буде та з них, для якої довжина вектора $\overrightarrow{P_1P_2} = (x_2 - x_1, y_2 - y_1)$ максимальна.

Аналогічно будемо шукати інші вершини. Припустимо, що вже знайдена i-та вершина $P_i(x_i, y_i)$ опуклої оболонки. Для наступної точки $P_{i+1}(x_{i+1}, y_{i+1})$ косі добутки $[\overrightarrow{P_iP_{i+1}}, \overrightarrow{P_iM}]$ невід'ємні для всіх точок M. Якщо таких точок декілька, то вибираємо ту, для якої вектор $\overrightarrow{P_iP_{i+1}}$ має найбільшу довжину. Пошук такої точки можна здійснювати так: спочатку ми можемо вважати наступною, *(i+1)*-шою, будь-яку точку. Потім знаходимо значення $[\overrightarrow{P_iP_{i+1}}, \overrightarrow{P_iM}]$, де M- всі інші точки. Якщо для однієї з них даний вираз буде менше нуля, то вважатимемо її наступною і продовжимо перевірку інших точок (аналогічно алгоритму пошуку мінімального елемента в масиві). Якщо значення виразу дорівнює нулю, то порівнюємо квадрати довжин векторів. В результаті за $O(N)$ операцію ми знайдемо наступну вершину опуклої оболонки. Продовжуючи процес далі, ми повернемося до точки P_1. А це означає, що опукла оболонка побудована.

При розв'язуванні даної задачі у випадку цілочисельних координат ми повністю можемо уникнути використання дійсної арифметики, а тому, уникнути від втрати точності обчислень. В противному випадку в розв'язках можуть бути отримані «лишні» точки, близькі до межі опуклої оболонки , або не враховані деякі з «потрібних» точок. Складність даного алгоритму

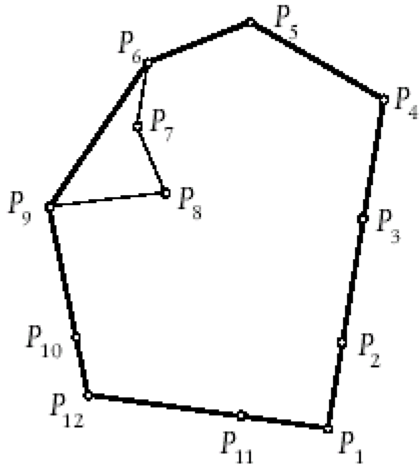
O(kN), де *k* – кількість точок опуклої оболонки, в гіршому випадку дорівнює *N*.

Наведемо варіант реалізації даного алгоритму. Множина даних точок знаходиться в масиві, а всі точки, що належать опуклій оболонці, будемо записувати в масив b.

```
type vv=record
          x, y: longint;
          end;
var a, b: array [1..100] of vv;
    min, m, i, j, k, n: integer;
function vect (a1, a2, b1,b2: vv): longint;
{косий добуток векторів a1a2 і b1b2}
begin
    vect:=(a2.x-a1.x)*(b2.y-b1.y)-(b2.x-b1.x)*(a2.y-a1.y)
end;
function dist2(a1, a2: vv): longint;
{квадрат довжини вектора  a1a2}
begin
    dist2:=sqr(a2.x-a1.x)+sqr(a2.y-a1.y)
end;
begin {Main}
    readln (n); {кількість точок}
for i:=1 to n do
read (a[i].x, a[i].y);
{шукаємо праву нижню точку}
m:=1;
for i:=2 to n do
    if a[i].y<a[m].y then m:=i else
    if (a[i].y=a[m].y) and (a[i].y>a[m].y) then m:=i;
{запишемо її в масив опуклої оболонки b і переставимо на перше місце в
масиві a}
b[1]:=a[m];
a[m]:=a[1];
a[1]:=b[1];
k:=1;
min:=2;
repeat
{шукаємо наступну вершину опуклої оболонки}
    for j:=2 to n do
      if (vect(b[k], a[min], b[k],a[j])<0 or ((vect(b[k], a[min], b[k],a[j])=0) and
      (dist2(b[k], a[min])< (dist2(b[k], a[j])))
      then min:=j;
```

```
k:=k+1;
{записана наступна вершина}
b[k]:=a[min];
min:=1;
until (b[k].x=b[1].x) and (b[k].y=b[1].y);
{поки ламана не замкнеться}
for j:=1 to k-1 do {друк результатів}
writeln (b[j].x,'',b[j].y)
end.
```

Існує інший алгоритм розв'язання цієї задачі (алгоритм Грехема) з обчислювальною складністю *O(NlogN)*, оснований на попередньому сортуванні точок даної множини за значенням кута в полярній системі координат з центром в одній із точок опуклої оболонки. Тобто найбільш складним буде сортування заданих точок. Сортування точок можна виконувати за знаком косого добутку $[\overrightarrow{P_1P_i}, \overrightarrow{P_1P_{i+1}}]$, де P_1 – будь-яка вершина опуклої оболонки (наприклад, права нижня точка). У відсортованому масиві точок всі дані добутки повинні бути невід'ємними. Точки з рівними кутами ($[\overrightarrow{P_1P_i}, \overrightarrow{P_1P_{i+1}}]=0$) розміщуються в порядку збільшення довжин відповідних векторів $\overrightarrow{P_1P_i}$ (мал. 19).



Мал.19. Вершини опуклої оболонки множини точок $\{P_i\}$: P_1, P_4, P_5, P_6, P_9, P_{12}. Номери всіх точок відповідають сортуванню за полярним кутом.

Далі перегляд Грехема використовує стек, в якому зберігаються точки, що є претендентами в опуклу оболонку. Спочатку в стек заносять першу з відсортованих точок. Потім – сусідню з нею вершину опуклої оболонки.

Якщо на першому із променів точок декілька, то ця точка променя P_i, найбільш віддалена від P_1. Третя – точка P_{i+1}. Нехай на вершині стека знаходиться точка P_k. Розглянемо наступну в порядку збільшення полярного кута точку заданої множини P_i. Поки частина ламаної P_{k-1}, P_k, P_i не є опуклим (див. задачу 4.2.), із стеку вилучається наступна точка P_k. Потім P_i заноситься до стеку. В момент закінчення перегляду всіх точок в стеці будуть знаходитись всі вершини опуклої оболонки. Так як будь-яка точка заноситься в стек не більше одного разу, тому час перегляду складає $O(N)$.

Наведемо приклад реалізації саме цього перегляду. Для наочності сортування зробимо методом «бульбашки». В якості стека використовується масив b. Описи змінних і функцій співпадають з наведеними вище.

```
begin
   readln (n);
   for i:=1 to n do
      read(a[i].x, a[i],y);
      {шукаємо праву нижню точку}
   m:=1;
   for i:=2 to n do
      if a[i].y< a[m].y then m:=i else
      if (a[i].y= a[m].y) and
        (a[i].x> a[m].x) then m:=i;
      {запишемо її в масив опуклої оболонки b  і переставимо на перше місце
в масиві a }
   b[1]:=a[m];
   a[m]:=a[1];
   a[1]:=b[1];
      {інші точки сортуємо методом «бульбашки» за значенням полярного
кута}
   for i:=n downto 3 do
     for j:=2 to i-1 do
     if (vect (a[1], a[j], a[1], a[j+1])<0 or
       ((vect (a[1], a[j], a[1], a[j+1])=0) and
       (dist2(a[1],a[j]>dist2(a[1], a[j+1])))
       then
     begin {b[n]-допоміжна змінна}
       b[n]:=a[j];
       a[j]:=a[j+1];
```

```
      a[j+1]:=b[n]
    end;
    {шукаємо другу вершину опуклої оболонки}
  i:=2;
  while vect(a[1], a[i+1], a[1], a[2])=0 do
    i:=i+1;
  b[2]:=a[i];
  b[3]:=a[i+1];
  k:=3;
  for i:=i+2 to n do
    begin
    while vect(b[k-1], b[k], b[k], a[i])<=0 do
    k:=k-1; {видаляємо точку з стека }
    k:=k+1;
    b[k]:=a[i] {додаємо точку в стек}
    end;
    for j:=1 to k do {друк розв'язку}
    writeln (b[j].x,'',b[j].y)
end.
```

На цьому ми закінчуємо огляд розв'язувань елементарних задач обчислювальної геометрії на площині. Їх можна використовувати при розв'язуванні більш складних, наприклад олімпіадних задач.

ЗАДАЧІ

Задача 1.

На площині задані дві точки A(x1,y1) і B(x2,y2). Визначити, який з відрізків - OA чи OB утворить більший кут з віссю OX.

Задача 2.

Чи належить точка площини А відрізку з кінцями В і С?

Задача 3.

1. Опуклий багатокутник задається координатами вершин при його обході за чи проти годинникової стрілки. Контур багатокутника не має самоперетинань. Визначити напрямок обходу.

2. Виконати те саме, але тільки у випадку не опуклого багатокутника.

Задача 4.

Відрізок на площині задається двома не співпадаючими точками X(x1,x2) і Y(y1,y2). З точки Z(z1,z2) до прямої, що містить відрізок [X,Y], проводиться перпендикуляр P.

Визначити, чи попадає перпендикуляр P на відрізок [X,Y] чи на його продовження.

Задача 5.

Багатокутник на площині задається координатами своїх N вершин у порядку обходу їх по контурі за годинниковою стрілкою. Вважається, що контур самоперетинань не має.

Знайти площу, периметр і кути багатокутника.

Задача 6.

Визначити, чи перетинаються пряма ax+b=y і відрізок з кінцями (x1,y1), (x2,y2).

Задача 7.

Два відрізки на площині задаються парами цілочисельних координат кінців. Визначити, чи перетинаються відрізки.

Задача 8.

1. Трикутник на площині задається цілочисельними координатами вершин. Для заданої точки Z(x,y) визначити, чи належить вона стороні трикутника чи лежить усередині чи поза ним.

2. Багатокутник на площині задається координатами своїх N вершин у порядку обходу їх по контурі за годинниковою стрілкою (контур самоперетинань не має). Для заданої точки Z(x,y) визначити, чи належить вона стороні багатокутника чи лежить всередині чи поза ним.

Задача 9.

На площині задані n відрізків координатами кінців. Кінці відрізків задаються двома парами координат (x1[i],y1[i]), (x2[i],y2[i]), 1<=i<=n (кінці належать відрізку).

Необхідно знайти пряму, що має спільні точки з максимальним числом відрізків, і надрукувати в порядку зростання номери тих відрізків, які ця пряма перетинає.

Задача 10.

N точок на площині задані своїми координатами. Знайти такий мінімальний по площі опуклий багатокутник, що всі N точок лежать або всередині цього багатокутника, або на його границі (такий опуклий багатокутник називається опуклою оболонкою).

Задача 11.

На сітці розміру m*n задано k точок своїми координатами. Необхідно визначити, чи можна побудувати опуклий багатокутник такий, що кожна точка належить деякій стороні.

Задача 12.

N точок на площині задані своїми координатами. Знайти порядок, у якому можна з'єднати ці точки, щоб вийшов N-кутник (тобто не було б перетинань сторін).

Задача 13.

Уявіть собі, що в зошиті Ви замалювали на аркуші деяку кількість клітинок і отримали клітинну фігуру.

Скільки осей симетрії має задана клітинна фігура.

Пояснення :

1) Задається S - число тестів. Для кожного тесту задаються NI розмір фігури по вертикалі, NJ - розмір фігури по горизонталі (NI<101, NJ<81) і сама фігура у вигляді NI рядків із пробілів і одиниць по NJ символів у кожному рядку.

2) Числа S, NI, NJ займають при введенні по три позиції.

Приклад .

Вхідні дані :

2 (кількість тестів)

2 (розмір 1-ої фігури по вертикалі)

4 (розмір 1-ої фігури по горизонталі)

1111

1 1

3 (розмір 2-ий фігури по вертикалі)

5 (розмір 2-ий фігури по горизонталі)

11111

111

111

Вихідні повідомлення:

1-А ФІГУРА МАЄ 1 ОСІ СИМЕТРІЇ

2-А ФІГУРА МАЄ 0 ОСІ СИМЕТРІЇ

Задача 14.

Прямокутник ABCD заданий координатами своїх вершин. На протилежних сторонах AB і CD задані послідовності R1 і R2 з N точок розбиття, а на сторонах BC і AD - R3 і R4 з M точок розбиття. Нумерація елементів послідовності R1 і R2 починається відповідно від точок A і D, а в

R3 і R4 – від B і A. З'єднавши відрізками точки з однаковими номерами в розбиттях R1 і R2, а потім у розбиттях R3 і R4, отримаємо розбиття Q прямокутника ABCD на множину чотирикутників. Побудувати алгоритм, що визначає чотирикутник розбиття Q з найбільшою площею, за умови, що відрізки, що з'єднують точки розбиття R1 і R2 паралельні стороні AD. Послідовності R1, R2, R3 і R4 задаються як масиви з довжин відрізків розбиття відповідних сторін прямокутника.

Задача 15.

На прямій задано N точок з координатами X1,X2,...,Xn. Написати програму, що знаходить на прямій таку точку Z, сума відстаней від якої до даних N точок мінімальна.

Задача 16.

На двовимірній площині задано N точок з координатами (X1,Y1), (X2,Y2), ..., (Xn,Yn). Написати програму, що знаходить таку точку Z(x,y), сума відстаней від якої до інших мінімальна і:

a) Z - одна з заданих точок;

b) Z - довільна точка площини.

Задача 17.

На площині розташовані N точок, задані своїми координатами. Знайти на осі абсцис точку, найбільша з відстаней від якої до обраних точок була б мінімальною.

Задача 18.

На площині задано N точок з координатами (X1,Y1), (X2,Y2), ..., (Xn,Yn). Написати програму, яка з цих точок виділяє вершини квадрата, що містить максимальне число заданих точок.

ПРИМІТКА: передбачається, що точки, розташовані на сторонах квадрата, належать йому.

Задача 19.

Задано на площині множину з N прямокутників, сторони яких паралельні осям координат, при цьому кожен прямокутник задається координатами лівої нижньої і правої верхньої його вершин.

Скласти алгоритм визначення найбільшого натурального числа К, для якого існує точка площини, що належить одночасно К прямокутникам.

Примітка: ефективним вважається алгоритм, число дій якого пропорційно N^2.

Задача 20.

На квадратному торті N свічок. Чи можна одним прямолінійним розрізом розділити його на дві рівні за площею частини, одна з яких не містила б ні однієї свічки? Свічки будемо вважати точками, у яких відомі їхній цілочисельні координати X[1], Y[1]; ...; X[N], Y[N] (початок координат - у центрі торта); розріз не може проходити через свічку.

Задача 21.

Дано N прямокутників (N>1), для яких передбачається, що:

А) Сторони будь-якого прямокутника паралельні координатним осям і прямокутник задається кінцями однієї з діагоналей.

В) Кожен прямокутник має спільні внутрішні точки хоча б з одним іншим прямокутником і не має спільних вершин, сторін чи частин сторін з жодним з інших прямокутників.

Скласти програму, що розв’язує наступні задачі:

1. За допомогою послідовності точок визначити зовнішній контур фігури F, що є об'єднанням прямокутників в замкнуту ламану. Приклад на малюнку.

2. Визначити чи містить фігура F "дірки" ,тобто замкнуті фігури, що їй не належать.

3. Розкласти фігуру на найменше можливе число не пересічних прямокутників, які не перетинаються, що можуть мати спільні сторони чи частини сторін , а їхнє об'єднання дає фігуру F.

4. Обчислити периметр і площу фігури F.

Примітка.

1. Задачі 3,4 розв’язується тільки для фігур, що не містять "дірки".

2. Повне розв’язання містить:

- аналіз (обґрунтування) розв’язку

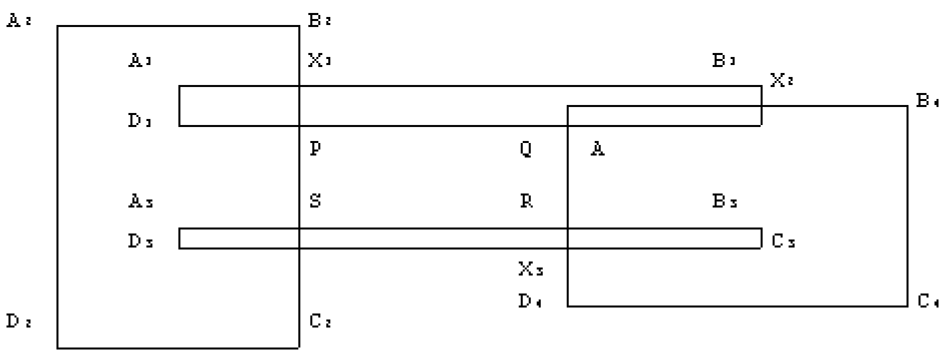
- текст програми з відповідними коментарями

- виконання текстового прикладу, що буде за даний при прийманні роботи

Об'єднання прямокутників Ai Bi Ci Di,i=1,2,3,4 є

фігура F=A2 B2 X1 B1 X2 B4 C4 D4 X3 X4 C2 D2

фігура F містить єдину "дірку" PQRS



Задача 22.

Контур міста.

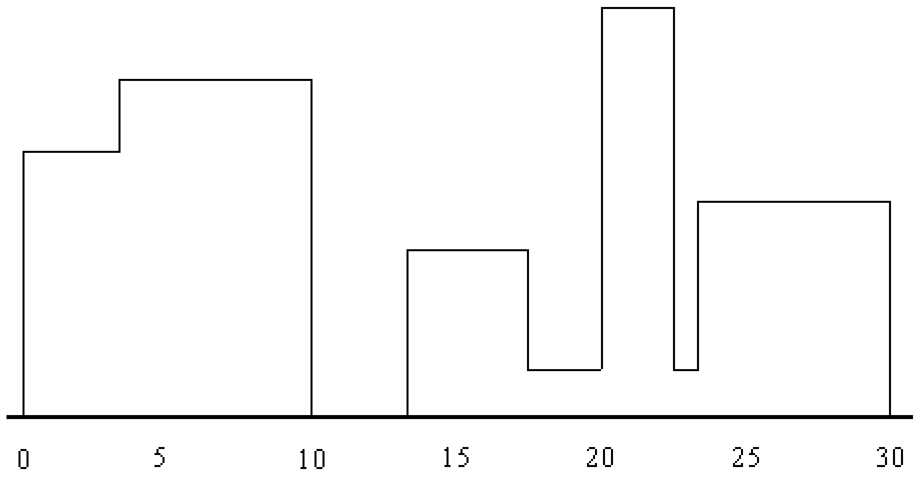
Необхідно написати програму - помічник архітектора в кресленні контуру міста. Місто задається розташуванням будинків. Місто розглядається як двовимірне і всі будинки в ньому - прямокутники, що мають однакову основу (місто побудоване на рівнині). Будинки задаються трійкою чисел (L[i],H[i],R[i]) де L[i] і R[i] є координати лівої і правої стін будинку i, а H[i] - висота цього будинку. На малюнку 1 будинки описуються трійками

(1,11,5), (2,6,7), (3,13,9), (12,7,16), (14,3,25), (19,18,22), (23,13,29),(24,4,28)

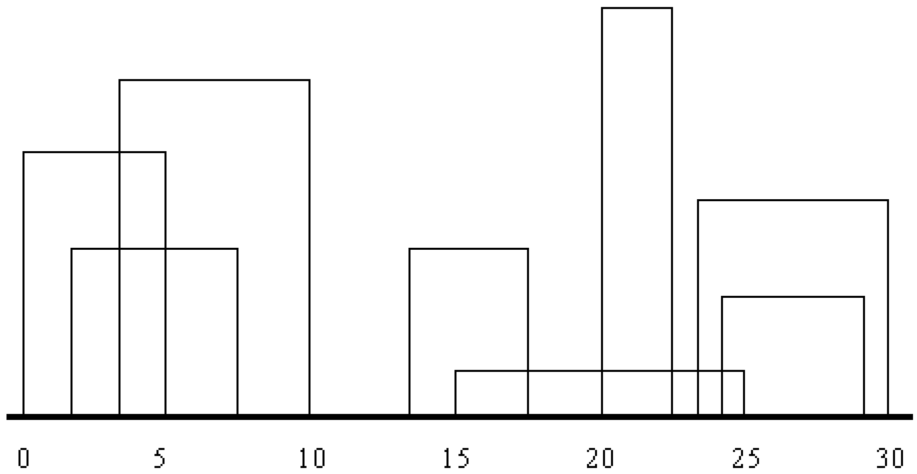
а контур, показаний на мал. 2, задається послідовністю

(1,11,3,13,9,0,12,7,16,3,19,18,22,3,23,13,29,0)

(про спосіб формування цієї послідовності див. нижче).



Мал. 1



Мал. 2

Введення.

Введення являє собою послідовність трійок, що задають будинки. Усі координати є цілі числа, менші 10000. В вхідному файлі мінімум один і максимум 50 будинків. Кожна трійка, що позначає будинок знаходиться в окремому рядку у вхідному файлі. Усі цілі числа в трійці розділені одним чи декількома пробілами. Трійки відсортовані по L[i], тобто по лівій x-координаті будинку, таким чином, будинок із самою мінімальною лівою x-координатою є першим у вхідному файлі.

Висновок.

Висновок буде складатися з вектора, що описує контур, як показано в прикладі вище. У векторі контуру (v[1],v[2],v[3], ... , v[n-2],v[n-1],v[n]), v[i], коли і-парне число, означає горизонтальну лінію (висоту). Коли і-непарне, v[i]- означає вертикальну лінію (х-координату). Вектор контуру буде визначати маршрут, пройдений, наприклад, жуком, що почав з мінімальної х-координаты і подорожує по усіх вертикальних і горизонтальних лініях, що визначає контур. Останній елемент у векторі лінії контуру буде 0.

Задача 23.

Нижня ліва і верхня права вершини прямокутника А відповідно мають координати (0,0) і (V,W). Множина S з N точок задається парами координат (x[i],y[i]), 1<=i<=N.

Знайти такий прямокутник G максимальної площі, що його сторони паралельні сторонам А, G цілком лежить у А (G і А можуть мати спільні граничні точки) і жодна точка з S не лежить усередині G (але може лежати на його стороні). Надрукувати значення площі G і координати нижньої лівої і верхній правої вершин цього прямокутника.

Якщо таких прямокутників декілька, то вивести інформацію по кожному.

Примітка: у множині S жодні дві точки не лежать на одній прямій, паралельній стороні А.

Необхідно:

1. Організувати введення даних у вигляді

< Введіть V і W- координату верхньої правої вершини прямокутника А >

< Введіть N - число точок >

< Введіть координати точок >

x[1]-> y[1]-i>

.....

x[N]-> y[N]-i>

2. Вивести результати у вигляді

< Максимальна площа прямокутника = >

< Координати нижньої лівої і верхній правої вершин прямокутника(ов) >

x1-> y1-> x2-> y2->

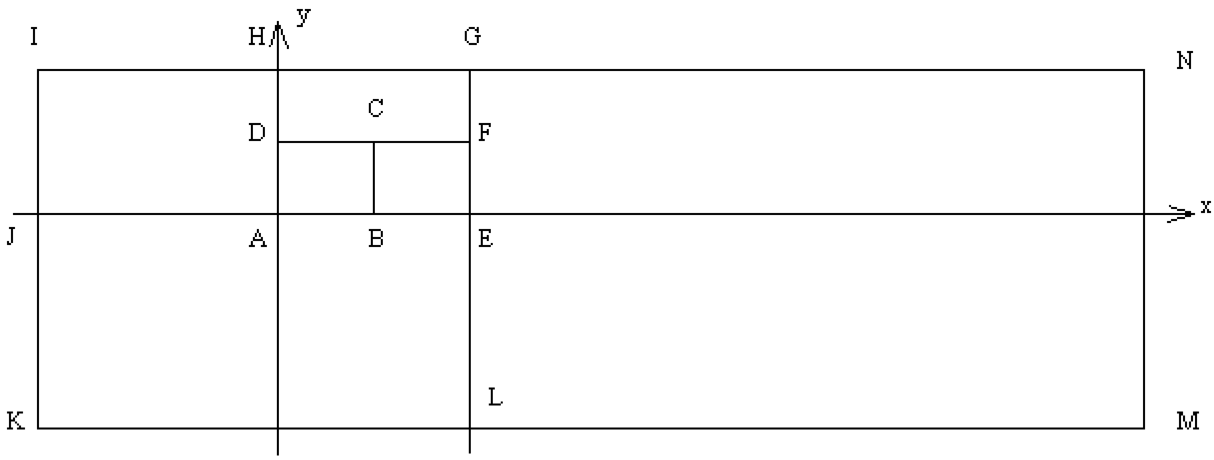
Задача 24.

У першій чверті координатної системи OXY намальований перший квадрат - ABCD, довжина сторони якого дорівнює 1 і вершина A знаходиться на початку координат. Потім зображені: другий квадрат - BEFC, третій - DFGH, четвертий - JAHI, п'ятий KLEJ, шостий - LMNG, і так далі «по спіралі» (мал.1).

Написати програму, що для введених цілих чисел x і y визначає і виводить номер квадрата, якому належить точка P(x;y). Якщо точка P лежить на сторонах квадратів чи у вершинах, то будемо вважати, що вона належить квадрату з найменшим номером з можливих.

Приклади:

X	Y	Результат
2	1	1
-1	0	4
13	2	10



Мал.1

Задача 25.

На площині N різних точок, що задані своїми координатами. Знайти рівняння прямої, що поділяє цю множину точок на 2 рівноцінних підмножини (тобто на підмножини з однаковою кількістю елементів).

Задача 26.

Знайти перетин й об'єднання двох опуклих багатокутників. Багатокутники задаються координатами вершин у порядку обходу по контурі.

Задача 27.

N-кутник на площині задається координатами вершин у порядку обходу по контурі (контур самоперетинів не має). Для точки Z(x,y) знайти мінімальну відстань до контуру N-кутника.

Задача 28.

На площині задано N точок своїми координатами і матриця C(N*N); C(i,j)=C(j,i)=1 у випадку, якщо вершини **i** і **j** з'єднані відрізком, інакше – 0. Відомо, що будь-яка вершина з'єднана принаймні з двома іншими, і що відрізки перетинаються тільки в кінцевих точках. Таким чином, вся площина розбивається на множину багатокутників. Задано точку Z(x,y). Знайти мінімальний за площею багатокутник, що містить Z, чи видати повідомлення, що такого не існує. Якщо Z належить деякому відрізку, то видати його кінці, якщо Z лежить у багатокутнику, то видати його вершини в порядку обходу по контурі.

Задача 29.

Будемо називати два багатокутники подібними, якщо існує взаємо-однозначне відображення сторін цих двох фігур таке, що відповідні сторони пропорційні з коефіцієнтом пропорційності k, а кути, утворені двома відповідними сторонами, рівні.

Визначити, чи подібні два багатокутники. Багатокутники задаються на площині координатами вершин контурів. Вершини в контурі перелічуються в порядку обходу проти годинникової стрілки.

Примітка: тому що всі обчислення на ЕОМ проводяться з обмеженою точністю, то вважати, що дві величини рівні якщо вони співпадають з точністю до двох знаків після коми.

ВВЕДЕННЯ: <з файлу T.TXT>

<Кількість вершин > N

<Багатокутник 1:>

<Координати вершини 1:> x11, y11

.....

<Координати вершини N:> x1N, y1N

<Багатокутник 2:>

<Координати вершини 1:> x21, y21

.....

<Координати вершини N:> x2N, y2N

ВИСНОВОК:

<Багатокутники не подібні>

чи

<Багатокутники подібні з k=> k

Задача 30.

Задано натуральне число N і дві послідовності цілих чисел (a1,a2,...,an) і (b1,b2,...,bn). Задані також два числа X0 і X1, X0<X1.

1. Знайти числа t(0), t(1), ..., t(p), p<=N, такі ,що X0=t(0)<t(1)<...<t(p)=X1 і вказати для кожного відрізка [t(j-1), t(j)], 1<=j<=p, таке число k, 1<=k<=N, що для всіх i, 1<=i<=N, і для всіх x з [t(j)-1,t(j)] справедлива нерівність ak*x+bk>=ai*x+bi.

2.Знайти числа s(0), s(1), ..., s(Q), такі, що X0=s(0)<s(1)<...<s(Q)=X1, і вказати для кожного відрізка [s(j-1),s(j)], 1<=j<=Q, таку перестановку (i1,i2,...,i) чисел 1,2,3,...,N, що для всіх X з [s(j-1),s(j)] справедливе нерівність ai1*x+bi1<=ai2*x+bi2<=...<=ai*x+bi і для всіх відрізків відповідні перестановки різні.

Задача 31.

На площині задана множина точок А и множина прямих В. Знайти дві такі різні точки з А, що пряма яка проходить через них паралельна найбільшій кількості прямих з В.

Задача 32.

У правильному n-кутнику провели кілька діагоналей, причому жодні три не перетинаються в одній точці. На скількі частин діагоналі розбили n-кутник? Діагоналі задані номерами вершин n-кутника, які вони сполучають, всі вершини пронумеровані по порядку числами 1, ...,n.

Задача 33.

Серед трикутників з вершинами в заданій множини точок на площині вказати такі, сторони якого містять максимальне число точок заданої множини.

Задача 34.

Усі стіни будинку мають довжину 5м. Північна та південна сторони – білі, західна і східні – сині. Людина пройшла від південно-східного кута будинку А метрів на південь, В метрів на схід і С метрів на північ і подивилася на будинок.

Написати алгоритм, який визначає, що бачить людина.

Задача 37.

На місцевості, що представляє собою ідеально рівну поверхню, стоїть високий пліт. План плоту являє собою замкнуту ламану без самоперетинань. Ця ламана задається N парами координат своїх вершин у порядку обходу області, що обмежується плотом, проти годинникової стрілки. Вершини пронумеровані від 1 до N, N<100.

У точці (x,y) стоїть людина ((x,y) не може лежати на ламаній). Вважаючи, що кожній ланці ламаної ставлять у відповідність пари номерів кінців, вказати, які ланки людина побачить повністю або частково в якості невирожденного відрізка, а яких - взагалі не побачить. Якщо при погляді ланку видно як точку чи як пару точок, то вважаємо, що іі не видно.

Задача 37.

На гранях двох рівних правильних тетраедрів N і M написані числа N1,N2,N3,N4 і M1,M2,M3,M4.

Чи можна сумістити тетраедри так, щоб на співпадаючих гранях виявилися однакові числа?

Використані джерела

1. Кормен Т., Аейзерсон Ч., Риверст Р. Алгоритмы. Построение и анализ. М.: МЦНМО, 2000.
2. Шикин Е.В., Боресков А.В., Зайцев А.А. Начала компьютерной графики. М.: Диалог-МИФИ, 1993.
3. Курант Р., Роббинс Г. Что такое математика. М.: ОГИЗ, государственное издательство технико-теоретической литературы, 1947.
4. Окулов С.М. 100 задач по информатике. Киров: Издательство ВГПУ, 2000.
5. Станкевич А.С. Решение задач I Всероссийской командной олимпиады по программированию. Информатика №12/2001.
6. Андреева Е.В. Геометрические задачи на олимпиадах по информатике. Информатика №14/2002.
7. Караванова Т.П. Основи алгоритмізації та програмування 750 задач з рекомендаціями та прикладами. Київ «Форум», 2002.
8. http://www.algolist.manual.ru
9. http://www.g6prog.narod.ru
10. http://www.uoi.kiev.ua