

## Zajęcia 26

**Temat:** Algorytmy geometryczne

**Czas trwania:** 2x45 min

**Cel zajęć:**

projektuje i programuje proste problemy z różnych dziedzin, stosuje przy tym: instrukcje wejścia/wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne, tablice, rekurencję, pisze własne funkcje rekurencyjne, struktury danych, biblioteka STL, algorytmy geometryczne: punkt, odcinek, prosta, wektor, wielokąt, testuje poprawność programów dla różnych danych, posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu, uruchamianiu i testowaniu programów;

**Efekty:**

- umie uruchomić potrzebne oprogramowanie,
- umie napisać program z wykorzystaniem algorytmów geometrycznych,
- zna iloczyn skalarny, wektorowy,
- zna wzór Picka.

**Formy i metody pracy:** praca samodzielna, omówienie, wykład

Zadania do wykonania na zajęciach	Treści programowe
1. Proste prostopadłe	M.6, P.2.18, P.2.20, A.3.13,
2. Fioletowe lasery	M.6, P.2.18, P.2.20, A.3.13,
3. Wiśniowy sad	M.6, P.2.18, P.2.20, A.3.13,

**Materiały do zajęć:**

[http://informatykaplus.edu.pl/upload/list/czytelnia/Przegląd\\_podstawowych\\_algorytmow.pdf](http://informatykaplus.edu.pl/upload/list/czytelnia/Przegląd_podstawowych_algorytmow.pdf)  
ss. 29-32

**Zadania do wykonania w domu:**

Radar (gotowa paczka do SIO2)

## ZADANIA I ROZWIĄZANIA

### Zadanie 1. Proste prostopadłe

Limit pamięci: 32MB

Dane są dwie proste, każda opisana za pomocą dwóch punktów. Mając dane współrzędne kartezjańskie punktów A, B (pierwsza prosta) oraz C i D (druga prosta) określ, czy te dwie proste są prostopadłe.

Wejście

Pierwszy wiersz danych zawiera liczbę naturalną  $W$  z zakresu od 1 do 1000, określającą ilość zestawów danych, czyli opisów par wektorów do wczytania. Każdy zestaw danych obejmuje dwa wiersze: w pierwszym znajdują się współrzędne pierwszej prostej (punkty A i B), a w drugim – współrzędne drugiej prostej (punkty C i D). Współrzędne punktów są liczbami całkowitymi, których wartość bezwzględna nie przekracza stu milionów. Liczby w wierszu oddzielone są od siebie pojedynczymi spacjami.

Wyjście

Program powinien wypisać dla każdego przypadku wiersz tekstu zawierający słowo TAK, jeśli dane wektory są prostopadłe, lub słowo NIE, jeśli nie są prostopadłe.

Przykład

Wejście	Wynik
2	TAK
0 0 1 1	NIE
2 0 1 1	
0 0 2 0	
0 0 2 1	

Rozwiązanie

Potraktujmy odcinki jako wektory i obliczmy ich współrzędne. W takim wypadku wystarczy sprawdzić, czy iloczyn skalarny tych wektorów  $(x_1 \cdot x_2 + y_1 \cdot y_2)$  jest równy 0.

### Zadanie 2. Fioletowe lasery

Dostępna pamięć: 256MB. Limit czasu: 1s

Bajtomir zajmuje się w Bajtocji wywiadem. Dostał właśnie nowe zadanie. Na podstawie meldunków szpiegów może wskazać położenie czujników ruchu w pewnym niedostępnym budynku. Dowiedział się też, że czujniki można zneutralizować specjalnym fioletowym laserem, ale trzeba to zrobić jednym strzałem. Bajtomir zastanawia się teraz, czy jest to możliwe. Pomóż mu!

Wejście

W pierwszej linii wejścia znajduje się jedna liczba całkowita  $n$  ( $1 \leq n \leq 10^6$ ), oznaczająca liczbę czujników ruchu w budynku. W kolejnych  $n$  liniach znajdują się po dwie liczby  $x_i$  oraz  $y_i$  ( $-10^6 \leq x_i, y_i \leq 10^6$ ) współrzędne  $i$ -tego czujnika ruchu.

Wyjście

Na wyjściu powinno znaleźć się jedno słowo: TAK lub NIE - odpowiedź na pytanie z zadania.

### Przykład

<b>Wejście</b> 3 0 0 -5 -5 5 5	<b>Wyjście</b> TAK
--	-----------------------

<b>Wejście</b> 3 0 0 1 2 2 1	<b>Wyjście</b> NIE
--	-----------------------

### Rozwiązanie

Zauważmy, że sprawdzamy, czy wszystkie punkty są współliniowe. Zapamiętujemy pierwszy punkt, szukamy kolejnego punktu o współrzędnych różnych niż punkt pierwszy. Następnie korzystając z iloczynu wektorowego sprawdzamy współliniowość punktów.

```
punkt a, b, c
wczytaj n
wczytaj a.x, a.y
i←2
wykonuj
  wczytaj b.x, b.y
  i ← i+1
dopóki (i≤n ∧ a.x=b.x ∧ a.y=b.y)
dopóki (i≤n)
  //jeżeli iloczyn wektorowy dla 3 punktów nie jest równy 0
  //punkty nie są współliniowe
  wczytaj c.x, c.y
  jeżeli ( (c.x-a.x) · (b.y-a.y) - (c.y-a.y) · (b.x-a.x) ≠ 0 )
    wypisz NIE i zakończ program
wypisz TAK
```

### Zadanie 3. Wiśniowy sad

Dostępna pamięć: 128 MB

Bajtomirek zakupił piękny wiśniowy sad. Sad jest piękny, ponieważ wszystkie drzewa posadzono obok siebie w odległości 1 zarówno w pionie, jak i w poziomie. Sad jest ogrodzony zwykłą siatką, a słupki, do których przymocowana jest siatka, są umieszczone w różnych miejscach, zawsze jednak o współrzędnych całkowitych względem położenia drzew. Żadne drzewo nie znajduje się na linii ogrodzenia. Bajtomirek zastanawia się teraz, ile dokładnie jest drzew wiśniowych w jego sadzie. Chciałby to szybko obliczyć, nie wie jednak jak. Pomóż mu!

Wejście

Pierwszy wiersz zawiera jedną liczbę całkowitą  $n$  – liczbę słupków ogrodzeniowych. Kolejnych  $n$  linii zawiera po dwie liczby całkowite  $x$  oraz  $y$  – współrzędne kolejnych słupków ( $3 \leq n \leq 10^3$ ,  $0 \leq x, y \leq 10^6$ ).

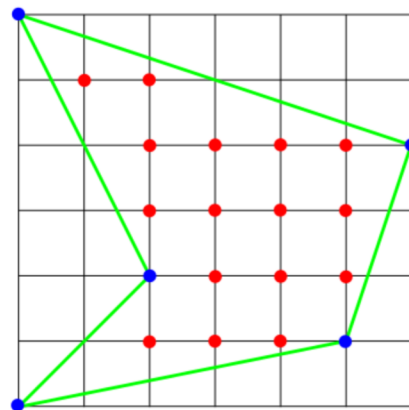
### Wyjście

Program powinien wypisać liczbę drzew wiśniowych znajdujących się wewnątrz ogrodzenia.

### Przykład

<p>Wejście:</p> <p>5</p> <p>0 0</p> <p>5 1</p> <p>6 4</p> <p>0 6</p> <p>2 2</p>	<p>Wyjście</p> <p>16</p>
---	--------------------------

Ilustracja przykładu:



### Rozwiązanie

W zadaniu rozwiązanie uzyskamy korzystając ze wzoru Picka:

$$P = W + \frac{1}{2}B - 1$$

Gdzie:  $P$  – pole powierzchni wielokąta,  $W$  – liczba punktów kratowych leżących wewnątrz wielokąta,  $B$  – liczba punktów kratowych leżących na brzegu wielokąta.

Musimy zatem wyznaczyć pole powierzchni i liczbę punktów kratowych leżących na brzegu wielokąta.

Iloczyn wektorowy wyznacza pole powierzchni równoległoboku. Pole powierzchni całkowitej wielokąta obliczymy sumując wszystkie pola trójkątów pomiędzy każdą parą sąsiadujących ze sobą wierzchołków wielokąta i punktu  $(0,0)$  (ostateczny wynik podzielimy na 2, możemy otrzymać ujemny).

Liczbę punktów kratowych leżących na brzegu wielokąta wyznaczymy z NWD współrzędnych  $x$  i  $y$  odcinków pomiędzy każdą parą sąsiadujących ze sobą wierzchołków wielokąta.

```
funkcja det(x1, y1, x2, y2)
    zwróć x1·y2-x2·y1

wczytaj n
dla i=0,1,2,...,n-1 wykonuj
    wczytaj x[i]. y[i]
x[n] ← x[0], y[n] ← y[0]
P ← 0, B ← 0
dla i=0,1,2,...,n-1 wykonuj
    P ← P det(x[i],y[i],x[i+1],y[i+1])
    B ← B + NWD( |x[i]-x[i+1]|, |y[i]-y[i+1]| )
P ← |P|
wypisz (P + 2 - B)/2
```