

**UZUPEŁNIA ZDAJĄCY**

**KOD**

--	--	--

**PESEL**

--	--	--	--	--	--	--	--	--	--	--	--

*miejsce  
na naklejkę*

**EGZAMIN MATURALNY Z INFORMATYKI**  
**POZIOM ROZSZERZONY**  
**CZĘŚĆ I**



MIN-R1\_1P-182

DATA: **11 maja 2018 r.**

GODZINA ROZPOCZĘCIA: **14:00**

CZAS PRACY: **60 minut**

LICZBA PUNKTÓW DO UZYSKANIA: **15**

**UZUPEŁNIA ZDAJĄCY**

**WYBRANE:**

.....  
(system operacyjny)

.....  
(program użytkowy)

.....  
(środowisko programistyczne)

**Instrukcja dla zdającego**

1. Sprawdź, czy arkusz egzaminacyjny zawiera 8 stron. Ewentualny brak zgłoś przewodniczącemu zespołu nadzorującego egzamin.
2. Rozwiązania i odpowiedzi zamieść w miejscu na to przeznaczonym.
3. Pisz czytelnie. Używaj długopisu/pióra tylko z czarnym tuszem/atramentem.
4. Nie używaj korektora, a błędne zapisy wyraźnie przekreśl.
5. Pamiętaj, że zapisy w brudnopisie nie podlegają ocenie.
6. Wpisz zadeklarowany (wybrany) przez Ciebie na egzamin system operacyjny, program użytkowy oraz środowisko programistyczne.
7. Jeżeli rozwiązaniem zadania lub jego części jest algorytm, to zapisz go w notacji wybranej przez siebie: listy kroków, pseudokodu lub języka programowania, który wybierasz na egzamin.
8. Na tej stronie oraz na karcie odpowiedzi wpisz swój numer PESEL i przyklej naklejkę z kodem.
9. Nie wpisuj żadnych znaków w części przeznaczonej dla egzaminatora.

**NOWA FORMUŁA**

Rozważamy następujący algorytm:

*Dane:*

$n$  – liczba całkowita dodatnia

*Wynik:*

$p$  – liczba całkowita dodatnia

$$p \leftarrow 1$$
$$q \leftarrow n$$

dopóki  $p < q$  wykonuj

$$s \leftarrow (p+q) \text{ div } 2$$

(\*) jeżeli  $s^*s^*s < n$  wykonaj

$$p \leftarrow s+1$$

w przeciwnym wypadku

$$q \leftarrow s$$

**Uwaga:** zapis **div** oznacza dzielenie całkowite.

### Zadanie 1.1. (0–3)

Podaj wynik działania algorytmu dla wskazanych w tabeli wartości  $n$ .

$n$	$p$
28	4
64	4
80	5

Miejsce na obliczenia.

[illegible]

Podaj najmniejszą oraz największą liczbę  $n$ , dla której wynikiem działania algorytmu będzie  $p = 10$ .

This image shows a full page of blank graph paper. The grid consists of small, equal-sized squares formed by thin black lines. There are 20 columns and 20 rows of squares, creating a total area of 400 small squares. The margins are consistent on all sides, and there are no markings, text, or drawings on the paper.

### Zadanie 1.3. (0–1)

☒ A. mniej niż  $2 \cdot \log_2 n$  razy.  
☐ B. więcej niż  $n/2$ , ale mniej niż  $n$  razy.  
☐ C. więcej niż  $n+1$ , ale mniej niż  $2n$  razy.  
☐ D. więcej niż  $n^2$  razy.

Wypełnia egzaminator	Nr zadania	1.1.	1.2.	1.3.
	Maks. liczba pkt.	3	2	1
	Uzyskana liczba pkt.			

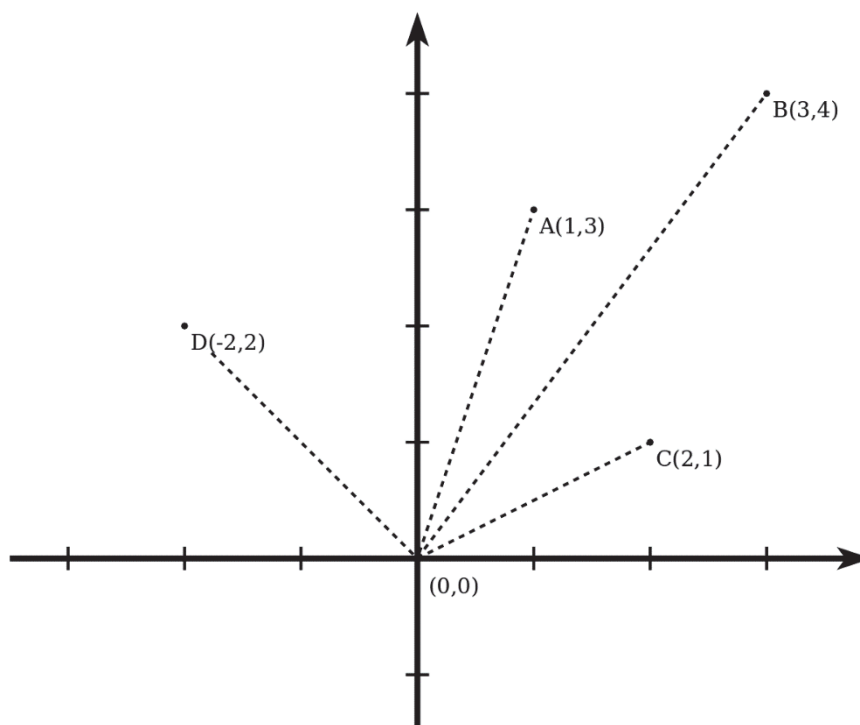
## Zadanie 2. Krajobraz

W pewnym paśmie górskim znajduje się  $n$  szczytów, które będziemy przedstawiać jako punkty w układzie kartezjańskim na płaszczyźnie. Wszystkie punkty leżą powyżej osi  $OX$ , tzn. druga współrzędna ( $y$ ) każdego punktu jest dodatnia.

W punkcie  $(0,0)$  stoi obserwator. Jeśli dwa szczyty  $A$  i  $B$  mają współrzędne  $(x_A, y_A)$  oraz  $(x_B, y_B)$ , to mówimy, że:

- szczyt  $A$  jest dla obserwatora *widoczny na lewo* od  $B$ , jeśli  $x_A/y_A < x_B/y_B$ ;
- szczyt  $B$  jest *widoczny na lewo* od  $A$ , jeśli  $x_A/y_A > x_B/y_B$ .

Wiemy, że żadne dwa szczyty nie leżą w jednej linii z obserwatorem, a zatem dla obserwatora te szczyty nie zasłaniają się nawzajem. Ilustrację przykładowego położenia szczytów można zobaczyć na poniższym rysunku:



W tym przykładzie, patrząc od lewej do prawej strony, obserwator widzi kolejno szczyt  $D$ , szczyt  $A$ , szczyt  $B$  i szczyt  $C$ .

Współrzędne szczytów dane są w dwóch tablicach  $X[1..n]$  oraz  $Y[1..n]$  – szczyt numer  $i$  ma współrzędne  $(X[i], Y[i])$ .

**Zadanie 2.1. (0–2)**

Napisz algorytm (w pseudokodzie lub wybranym języku programowania), który znajdzie i poda współrzędne *skrajnie lewego szczytu*, tzn. widocznego dla obserwatora na lewo od wszystkich pozostałych szczytów.

**Specyfikacja:**

*Dane:*

$n$  – liczba całkowita dodatnia

$X[1..n]$  – tablica liczb całkowitych

$Y[1..n]$  – tablica liczb całkowitych dodatnich

Para  $(X[i], Y[i])$  to współrzędne jednego szczytu,  $i = 1, 2, \dots, n$ .

Żadne dwa szczyty nie leżą w jednej linii z obserwatorem.

*Wynik:*

$x, y$  – współrzędne skrajnie lewego szczytu spośród tych opisanych w tablicach  $X$  i  $Y$ .

Algorytm



Wypełnia egzaminator	Nr zadania	2.1.
	Maks. liczba pkt.	2
	Uzyskana liczba pkt.	

### Zadanie 2.2. (0–4)

Napisz algorytm (w pseudokodzie lub wybranym języku programowania), który przestawi elementy tablic  $X$  i  $Y$  tak, aby szczyty były uporządkowane w kolejności, w której obserwator widzi je od lewej do prawej strony. Aby otrzymać maksymalną ocenę, Twój algorytm powinien mieć złożoność czasową kwadratową lub mniejszą.

Algorytm może używać wyłącznie instrukcji sterujących, operatorów arytmetycznych, operatorów logicznych, porównań i przypisań do zmiennych. Zabronione jest używanie funkcji bibliotecznych dostępnych w językach programowania.

#### Specyfikacja:

*Dane:*

$n$  – liczba całkowita dodatnia

$X[1..n]$  – tablica liczb całkowitych

$Y[1..n]$  – tablica liczb całkowitych dodatnich

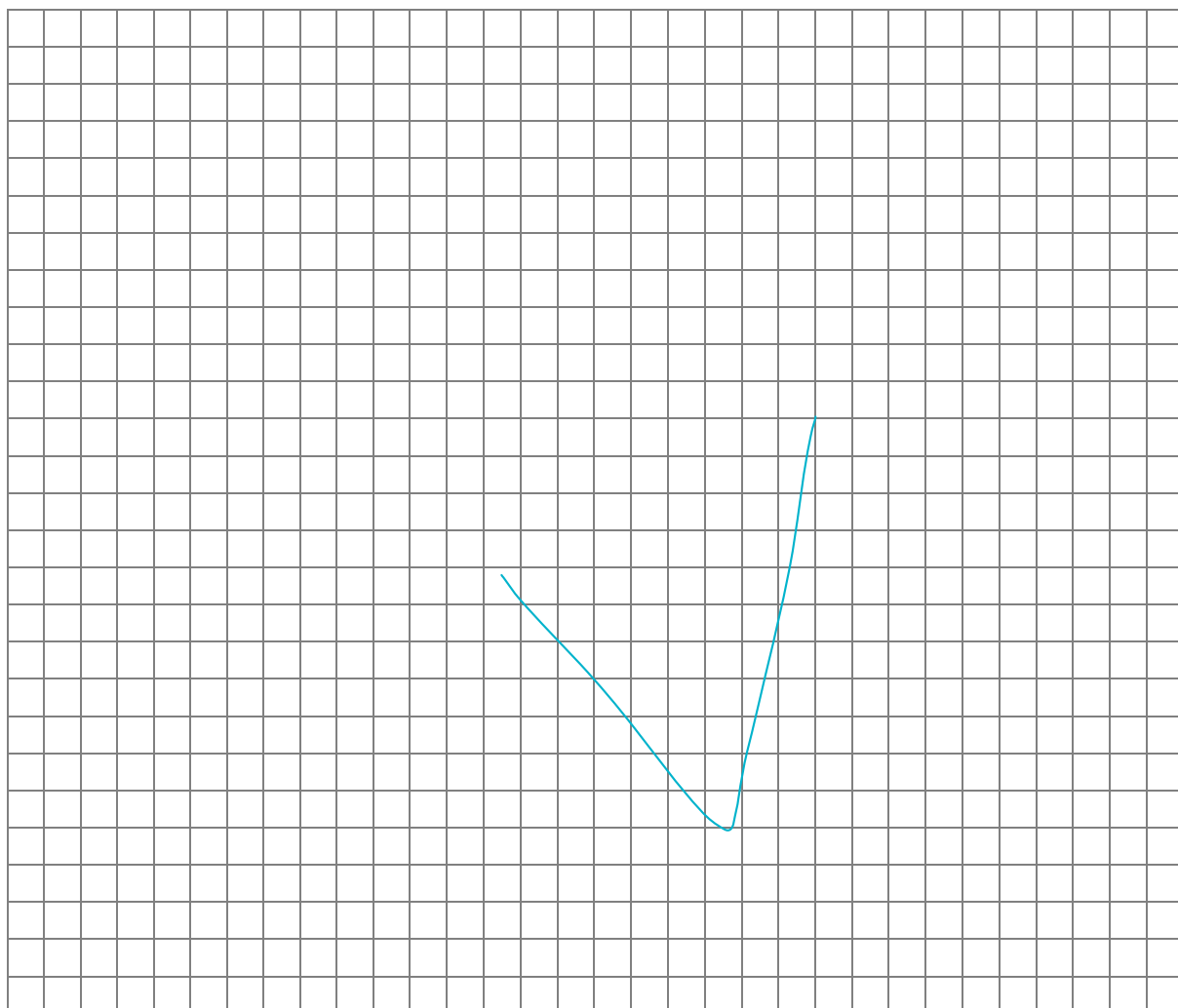
Para  $(X[i], Y[i])$  to współrzędne jednego szczytu,  $i = 1, 2, \dots, n$ .

Żadne dwa szczyty nie leżą w jednej linii z obserwatorem.

*Wynik:*

$X[1..n], Y[1..n]$  – tablice zawierające współrzędne danych szczytów, uporządkowanych w kolejności, w której obserwator widzi je od lewej do prawej strony.

Algorytm



**Zadanie 3. Test**

Oceń prawdziwość podanych zdań. Zaznacz P, jeśli zdanie jest prawdziwe, albo F – jeśli jest fałszywe.

W każdym zadaniu punkt uzyskasz tylko za komplet poprawnych odpowiedzi.

**Zadanie 3.1. (0–1)**

Na pewnym serwerze WWW znajduje się strona napisana w języku PHP, a jej kod zawiera fragmenty w języku JavaScript. Pewien komputer-klient pobrał i wyświetlił tę stronę. Wiadomo, że:

1.	kod PHP jest wykonywany przez komputer – serwer.	<u>P</u>	F
2.	kod JavaScript jest wykonywany przez komputer – klient.	<u>P</u>	F
3.	podczas wykonywania kodu PHP zawsze pobierane są dane od klienta.	P	<u>F</u>
4.	podczas wykonywania kodu JavaScript mogą być pobierane dodatkowe dane zarówno od klienta, jak i od serwera.	<u>P</u>	F

**Zadanie 3.2. (0–1)**

1.	Plakat do druku lepiej przygotować w modelu barw RGB niż CMYK.	P	<u>F</u>
2.	Kolor żółty jest kolorem podstawowym w modelu RGB.	P	<u>F</u>
3.	W wyniku nałożenia się składowych Yellow i Magenta w modelu CMYK otrzymamy kolor czerwony.	<u>P</u>	F
4.	W modelu barw CMYK litera C pochodzi od angielskiego słowa <i>contrast</i> .	P	<u>F</u>

**Zadanie 3.3. (0–1)**

Wskaż zdania prawdziwe dla języka SQL.

1.	W wynikach zapytania postaci <code>SELECT (...) ORDER BY (...)</code> zawsze dostajemy rekordy uporządkowane ściśle rosnąco według wskazanego pola.	P	<u>F</u>
2.	Zapytanie <code>UPDATE</code> może zmienić wartości pól w bazie danych.	<u>P</u>	F
3.	Zapytanie postaci <code>SELECT * FROM tabela1 WHERE pole LIKE (...)</code> może w pewnych warunkach dać wszystkie rekordy z tabeli <i>tabela1</i> .	<u>P</u>	F
4.	Wynik zapytania <code>SELECT * FROM tabela1 JOIN tabela2 ON tabela1.pole = tabela2.pole</code> może być pusty przy niepustych tabelach <i>tabela1</i> oraz <i>tabela2</i> .	<u>P</u>	F

Wypełnia egzaminator	Nr zadania	2.2.	3.1.	3.2.	3.3.
	Maks. liczba pkt.	4	1	1	1
	Uzyskana liczba pkt.				

## **BRUDNOPIS (*nie podlega ocenie*)**