

UZUPEŁNIA ZDAJĄCY

KOD

--	--	--

PESEL

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

*Miejsce
na naklejkę
z kodem*

**EGZAMIN MATURALNY
Z INFORMATYKI**

POZIOM ROZSZERZONY

CZĘŚĆ I

13 MAJA 2019

**Godzina rozpoczęcia:
14:00**

Instrukcja dla zdającego

1. Sprawdź, czy arkusz egzaminacyjny zawiera 9 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 obok zadeklarowane (wybrane) przez Ciebie na egzamin środowisko komputerowe, kompilator języka programowania oraz program użytkowy.
7. Jeżeli rozwiązaniem zadania lub jego części jest algorytm, to zapisz go w wybranej przez siebie notacji: listy kroków, schematu blokowego lub języka programowania, który wybrałeś/aś 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.

WYBRANE:

.....
(środowisko)

.....
(kompilator)

.....
(program użytkowy)

**Czas pracy:
90 minut**

**Liczba punktów
do uzyskania: 20**



MIN-R1_1P-192

Zadanie 1. Dwie tablice

Przeanalizuj poniższy algorytm, który dla dodatniej liczby całkowitej n i tablicy liczb całkowitych $A[1..n]$ oblicza inną tablicę $C[1..n]$:

dla $i = 1, 2, \dots, n$:

$C[i] = 1$

dla $j = 1, 2, \dots, i-1$:

jeśli $A[j] < A[i]$ **oraz** $C[j]+1 > C[i]$:

$C[i] = C[j]+1$

Zadanie 1.1. (4 pkt)

Uzupełnij tabelę. Podaj zawartość tablicy C po wykonaniu powyższego algorytmu.

Tablica A	Tablica C
[4,2,3,1,5]	[1,1,2,1,3]
[3,3,3,3]	[1,1,1,1]
[1,2,3,4]	
[1,3,2,5]	
[3,5,1,4,6,7,2,10,9,11]	
[3,1,4,2,6,5,7]	

Miejsce na obliczenia:

--

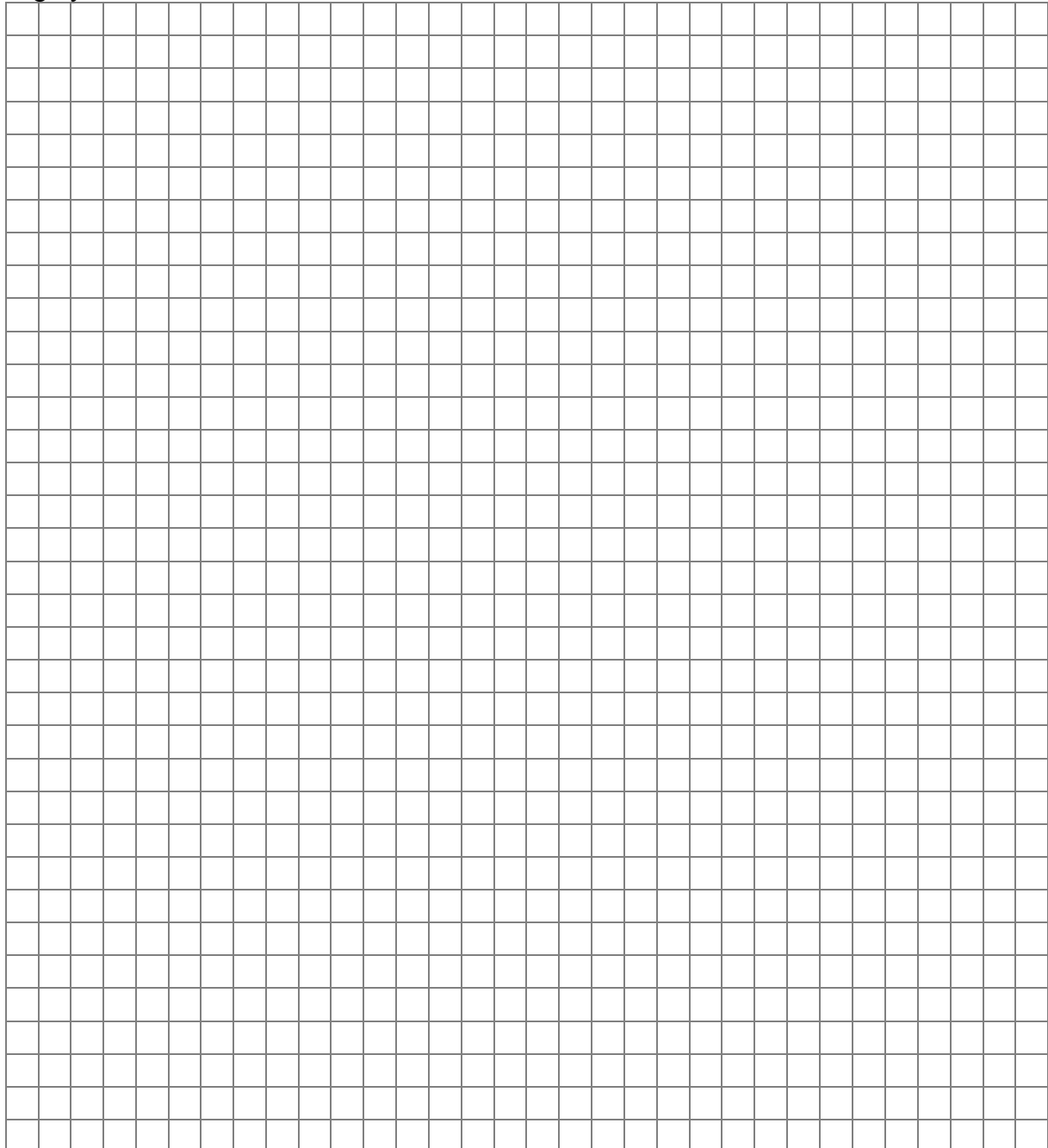
Zadanie 2. Test pierwszości

Zadanie 2.1. (2 pkt)

W wybranej przez siebie notacji (schemat blokowy, pseudokod, lista kroków, język programowania) napisz funkcję $\text{pot}(a, k)$, której wynikiem jest $a^k \bmod k$, gdzie a i k to liczby naturalne i $2 \leq a < k$. Operacja \bmod oznacza resztę z dzielenia całkowitego.

Uwaga: W zapisie możesz wykorzystać tylko operacje dodawania, odejmowania, mnożenia, dzielenia, dzielenia całkowitego, reszty z dzielenia lub samodzielnie napisane funkcje.

Algorytm:



Zadanie 2.2. (2 pkt)

Liczba pierwsza to liczba całkowita większa od 1, która ma dokładnie dwa dzielniki naturalne: jedynkę i siebie samą.

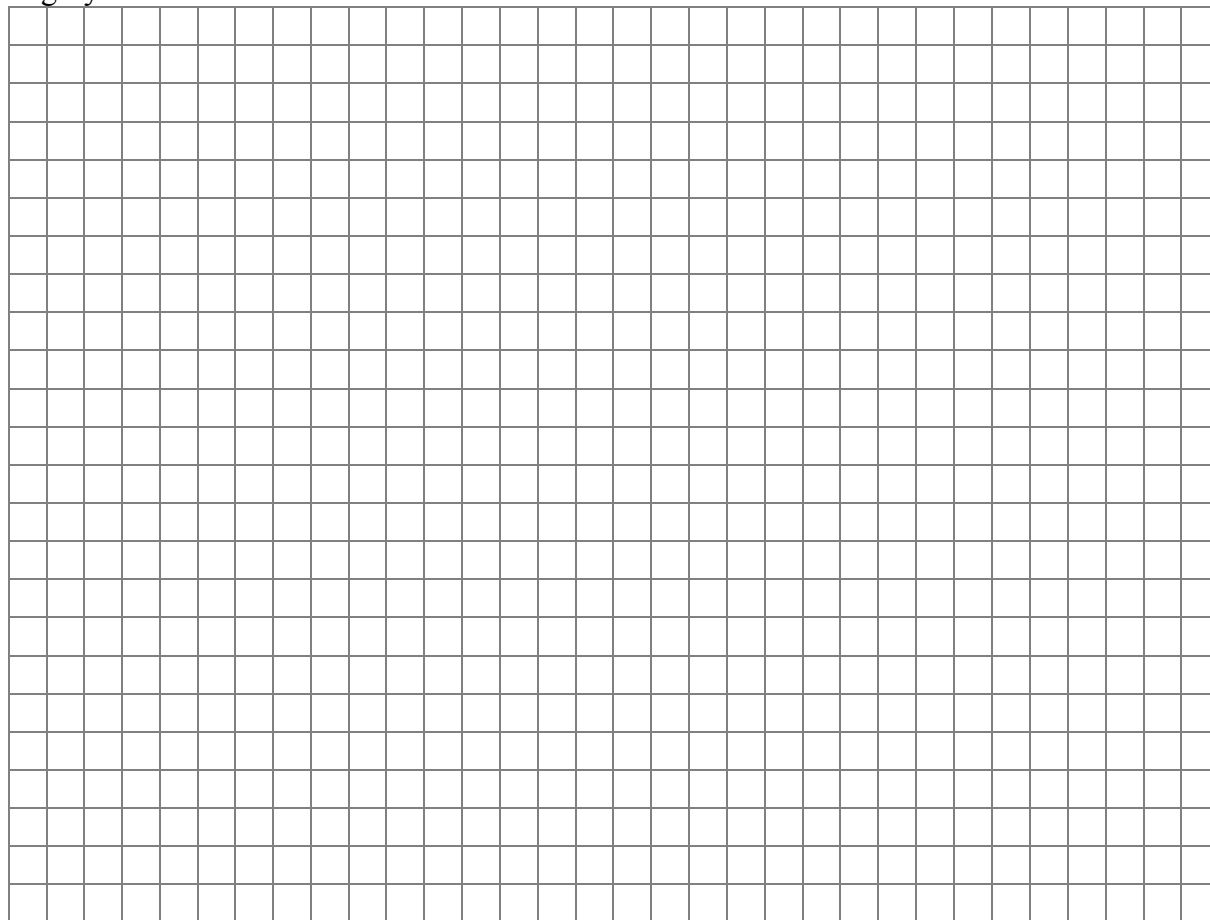
Test pierwszości Fermata polega na sprawdzeniu, czy dana liczba naturalna jest *prawdopodobnie pierwsza*.

Liczba całkowita $k > 2$ jest *prawdopodobnie pierwsza*, jeżeli dla każdego całkowitego a , gdzie $2 \leq a < k$, spełniony jest warunek: $a^k \bmod k = a$.

W wybranej przez siebie notacji (schemat blokowy, pseudokod, lista kroków, język programowania) napisz funkcję `testF(k)` (gdzie k to liczba całkowita większa od 2), której wynikiem jest `1`, gdy liczba k jest *prawdopodobnie pierwsza*, a `0` gdy k nie jest *prawdopodobnie pierwsza*.

Uwaga: W zapisie możesz wykorzystać tylko operacje dodawania, odejmowania, mnożenia, dzielenia, dzielenia całkowitego, reszty z dzielenia, samodzielnie napisane funkcje oraz funkcję `pot(a, k)` opisaną w zadaniu 2.1.

Algorytm:



Wypełnia egzaminator	Nr zadania	2.1.	2.2.
	Maks. liczba pkt.	2	2
	Uzyskana liczba pkt.		

Zadanie 3.3. (1 pkt)

1.	Adres IPv6 składa się z 64 bitów.	P	F
2.	Adres IPv6 składa się z 128 bitów.	P	F
3.	Adres IPv4 składa się z 64 bitów.	P	F
4.	Adres IPv4 składa się z 32 bitów.	P	F

Zadanie 3.4. (1 pkt)

Skrótem nazwy złącza, przez które można podłączyć urządzenia peryferyjne do komputera, jest

1.	USB	P	F
2.	FTP	P	F
3.	PHP	P	F
4.	HDMI	P	F

Wypełnia egzaminator	Nr zadania	3.3.	3.4.
	Maks. liczba pkt.	1	1
	Uzyskana liczba pkt.		

BRUDNOPIS (*nie podlega ocenie*)