

WYPEŁNIA ZDAJĄCY

KOD

--	--	--

PESEL

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

Miejsce na naklejkę.

Sprawdź, czy kod na naklejce to
E-100.

Jeżeli tak – przyklej naklejkę.
Jeżeli nie – zgłoś to nauczycielowi.

Egzamin maturalny

Formuła 2015

INFORMATYKA

Poziom rozszerzony

Część I

WYPEŁNIA ZDAJĄCY

WYBRANE:

.....
(system operacyjny)

.....
(program użytkowy)

.....
(środowisko programistyczne)

Symbol arkusza

EINP-R1-100-2305

DATA: **22 maja 2023 r.**

GODZINA ROZPOCZĘCIA: **9:00**

CZAS TRWANIA: **60 minut**

LICZBA PUNKTÓW DO UZYSKANIA: **15**

Przed rozpoczęciem pracy z arkuszem egzaminacyjnym

1. Sprawdź, czy nauczyciel przekazał Ci **właściwy arkusz egzaminacyjny**, tj. arkusz we **właściwej formule**, z **właściwego przedmiotu** na **właściwym poziomie**.
2. Jeżeli przekazano Ci **niewłaściwy** arkusz – natychmiast zgłoś to nauczycielowi. Nie rozrywaj banderol.
3. Jeżeli przekazano Ci **właściwy** arkusz – rozerwij banderole po otrzymaniu takiego polecenia od nauczyciela. Zapoznaj się z instrukcją na stronie 2.



Instrukcja dla zdającego

1. Sprawdź, czy arkusz egzaminacyjny zawiera 11 stron (zadania 1–3). Ewentualny brak zgłoś przewodniczącemu zespołu nadzorującego egzamin.
2. Na pierwszej stronie oraz na karcie odpowiedzi wpisz swój numer PESEL i przyklej naklejkę z kodem.
3. Wpisz zadeklarowane (wybrane) przez Ciebie na egzamin: system operacyjny, program użytkowy oraz środowisko programistyczne.
4. Odpowiedzi i rozwiązania zapisz w miejscu na to przeznaczonym przy każdym zadaniu.
5. Pisz czytelnie. Używaj długopisu/pióra tylko z czarnym tuszem/atramentem.
6. Nie używaj korektora, a błędne zapisy wyraźnie przekreśl.
7. Pamiętaj, że zapisy w brudnopisie nie będą oceniane.
8. Nie wpisuj żadnych znaków w części przeznaczonej dla egzaminatora.
9. Możesz korzystać z kalkulatora prostego.

**Zadania egzaminacyjne są wydrukowane
na następnych stronach.**

Zadanie 1. Ciąg i jego opis

Rozważamy ciągi złożone z liczb całkowitych dodatnich. Jeżeli mamy pewien ciąg A , możemy skonstruować drugi ciąg B , będący opisem A , w następujący sposób: każdy fragment A będący p -krotnym powtórzeniem jednej liczby x zamieniamy na dwie liczby p i x w ciągu B .

Przykład: ciąg (1, 1, 3, 2, 2, 2, 1) to „dwie jedynki, jedna trójka, trzy dwójki i jedna jedynka”, a więc jego opis to (2, 1, 1, 3, 3, 2, 1, 1).

Z kolei ciąg (2, 2, 2, 2, 5, 3, 3) to kolejno „cztery dwójki, jedna piątka, dwie trójki”, więc jego opis to (4, 2, 1, 5, 2, 3).

Zadanie 1.1. (0–2)

Uzupełnij poniższą tabelę – wpisz w odpowiednie pola: opisy dla podanych przykładów ciągu A , długości tych opisów oraz ciąg A , dla którego podano opis i długość tego opisu.

Ciąg A	Opis ciągu A (ciąg B)	Długość opisu ciągu A (liczba elementów ciągu B)
(1, 1, 3, 2, 2, 2, 1)	(2, 1, 1, 3, 3, 2, 1, 1)	8
(3, 3, 3, 2, 2, 1, 1, 1, 6)		
(2, 2, 2, 2, 2, 2)		
	(4, 1, 1, 4)	4

Zadanie 1.2. (0–3)

Zapisz w pseudokodzie lub wybranym języku programowania algorytm, który dla danego ciągu A , zapisanego w tablicy $A[1..n]$, obliczy długość jego opisu (liczbę elementów ciągu B) zgodnie z podanymi wcześniej regułami.

Uwaga: W zapisie możesz wykorzystać tylko operacje arytmetyczne (dodawanie, odejmowanie, mnożenie, dzielenie, dzielenie całkowite, reszta z dzielenia), porównywanie znaków i liczb, odwoływanie się do pojedynczego elementu tablicy, instrukcje sterujące, przypisania do zmiennych lub samodzielnie napisane funkcje, wykorzystujące wyżej wymienione operacje. **Zabronione** jest używanie funkcji wbudowanych oraz operatorów innych niż wymienione, dostępnych w językach programowania.

Specyfikacja:

Dane:

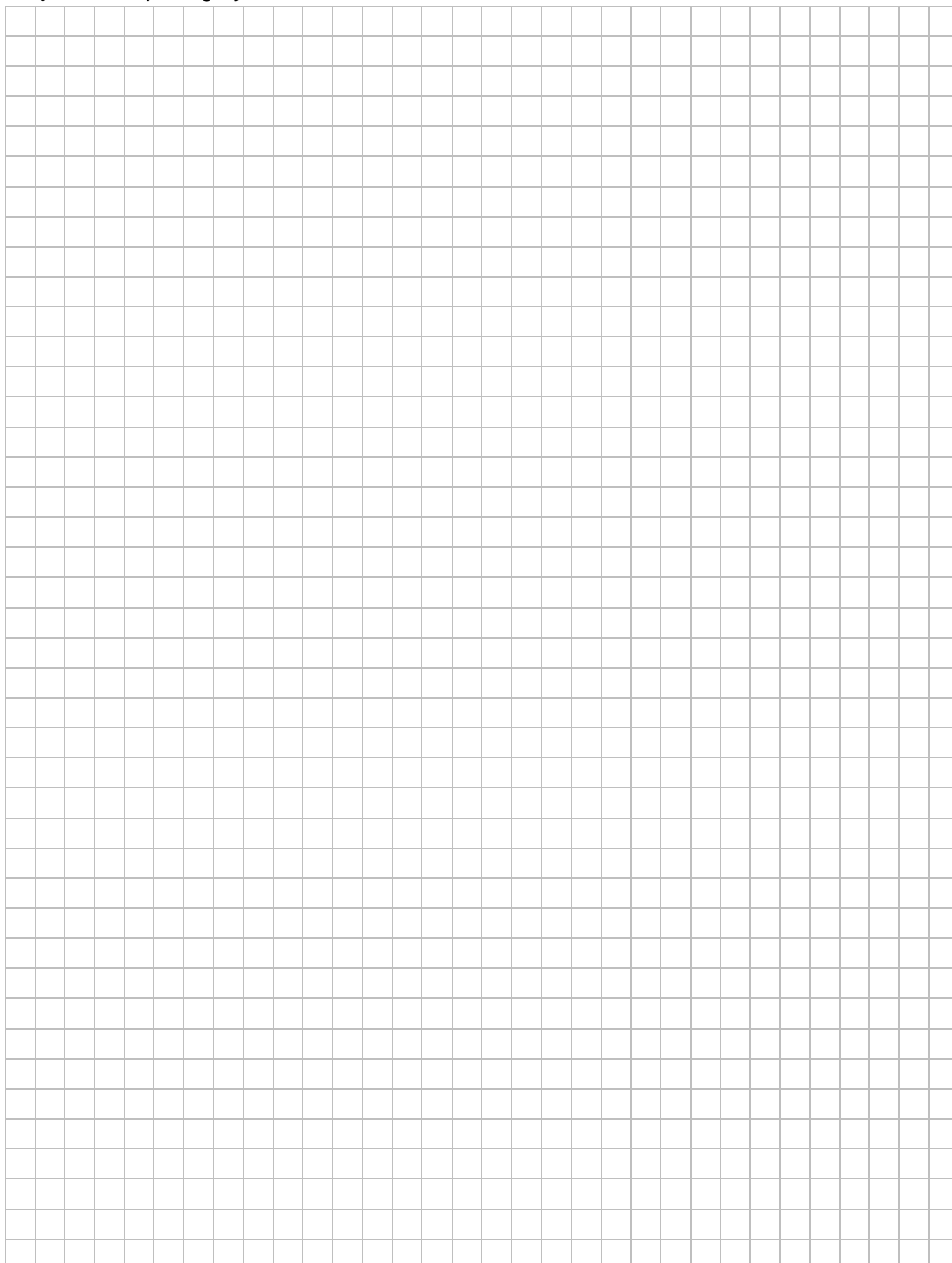
n – liczba całkowita dodatnia – liczba elementów ciągu A

$A[1..n]$ – tablica liczb całkowitych dodatnich zawierająca kolejne elementy ciągu A

Wynik:

w – liczba elementów ciągu B , będącego opisem ciągu A

Miejsce na zapis algorytmu



Więcej arkuszy znajdziesz na stronie: arkusze.pl

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

Zadanie 2. Funkcja rekurencyjna

Dane są dodatnia liczba całkowita n i tablica $A[1..n]$ zawierająca n dodatnich liczb całkowitych. Przeanalizuj działanie zapisanej poniżej funkcji f , której parametry p i q spełniają warunek $1 \leq p \leq q \leq n$.

$f(p, q)$:

jeżeli $p \neq q$
 $k \leftarrow (q - p + 1) \operatorname{div} 2$
dla $i = 1, 2, \dots, k$
 zamień($A[p + i - 1]$, $A[q - k + i]$)
 $f(p, p + k - 1)$
 $f(q - k + 1, q)$

Uwaga:

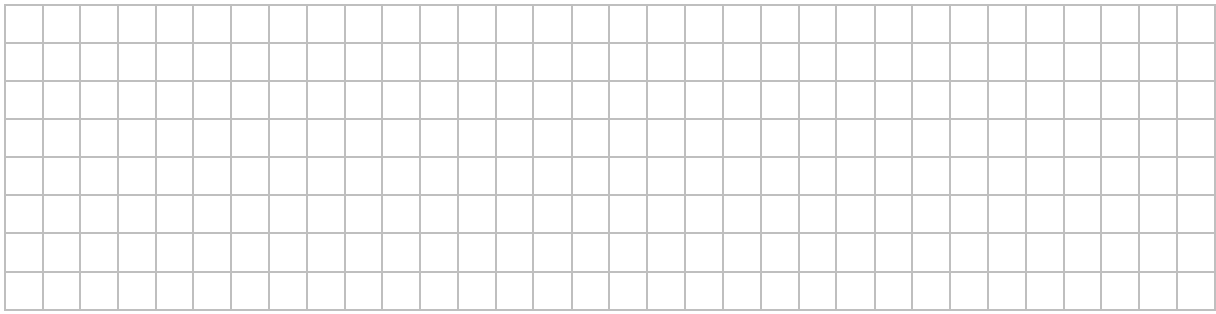
- div jest operatorem oznaczającym część całkowitą z dzielenia
- operacja $\operatorname{zamień}(x, y)$ zamienia ze sobą wartości zmiennych x i y
- \leftarrow jest operatorem przypisania; $x \leftarrow 2$ oznacza, że wartość x staje się 2

Zadanie 2.1. (0–3)

Uzupełnij tabelę – dla podanych wartości n i tablicy $A[1..n]$ podaj, jaki będzie stan tablicy A (jakie będą wartości elementów tej tablicy) po zakończeniu działania funkcji f wywołanej z parametrami $1, n$.

n	A	Zawartość A po wykonaniu $f(1, n)$
3	[2, 3, 4]	
4	[1, 2, 3, 4]	
10	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]	

Miejsce na obliczenia



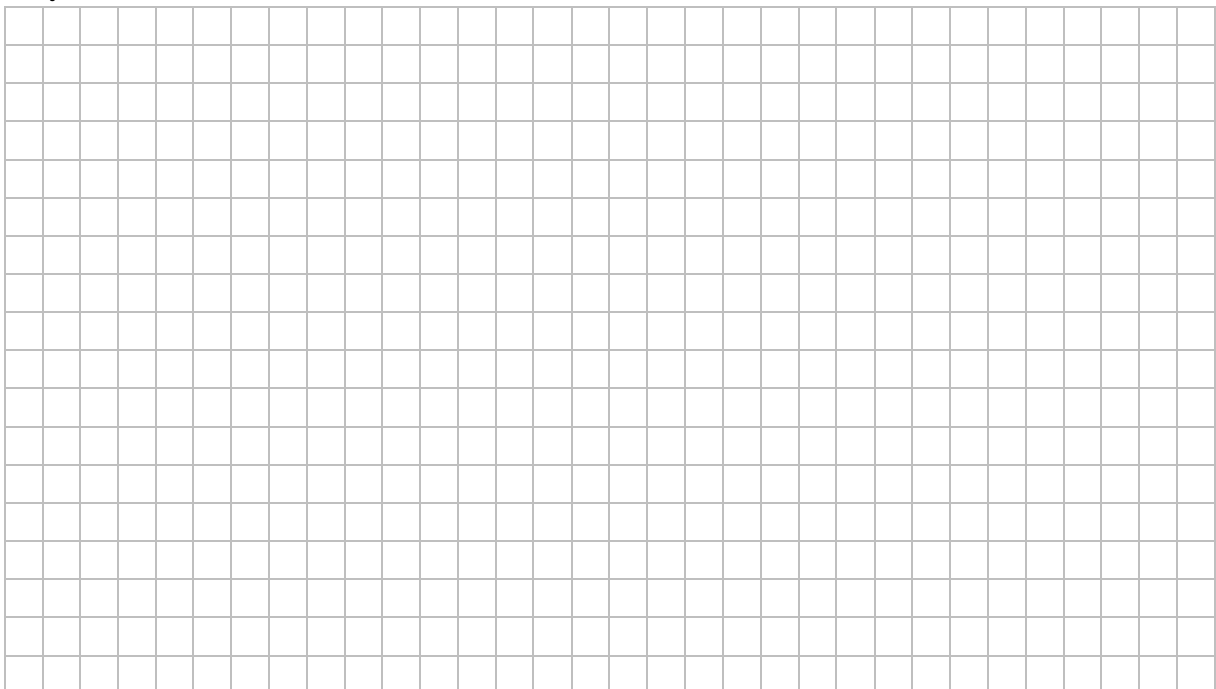
Zadanie 2.2. (0–3)

Uzupełnij tabelę – podaj, ile razy po wywołaniu $f(1, n)$ dla tablicy $A[1..n]$ wykonana zostanie operacja *zamień()*.

Uwaga: zauważ, że liczba wykonań operacji *zamień* nie zależy od zawartości tablicy A , a tylko – od jej długości.

n	Liczba operacji <i>zamień()</i> po wywołaniu $f(1, n)$
4	4
8	
16	
256	

Miejsce na obliczenia



Wypełnia egzaminator	Nr zadania	2.1.	2.2.
	Maks. liczba pkt	3	3
	Uzyskana liczba pkt		

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)

Rozważamy dwie funkcje **F** i **G**, których argumentem jest liczba całkowita $x > 1$ (gdzie mod oznacza resztę z dzielenia):

F(x):

$i \leftarrow 2$

dopóki $x \bmod i \neq 0$ **wykonuj**

$i \leftarrow i + 1$

zwróć i

G(x):

$i \leftarrow x - 1$

dopóki $x \bmod i \neq 0$ **wykonuj**

$i \leftarrow i - 1$

zwróć i

1.	$F(2)=2$ oraz $G(2)=1$.	P	F
2.	Dla każdej liczby parzystej x wartość $F(x)$ jest parzysta.	P	F
3.	Dla każdej liczby parzystej x wartość $G(x)$ jest parzysta.	P	F
4.	Dla każdej liczby x większej od 2 $F(x)$ dzieli liczbę x .	P	F

Zadanie 3.2. (0–1)

1.	$A_{5_{16}} = 245_8$	P	F
2.	$A_{5_{16}} < 10100100_2$	P	F
3.	$10100100_2 = 2210_4$	P	F
4.	$2210_4 < 245_8$	P	F

Informacja do zadań 3.3. i 3.4.

W bazie danych znajdują się tabele *zwierzeta* oraz *gromady*.

Tabela *zwierzeta* zawiera dane: identyfikator zwierzęcia (*id*), gatunek (*gatunek*), sposób odżywiania (*sposob_odzywiania* – mięsożerne, roślinożerne) oraz identyfikator gromady, do której należy dany gatunek (*id_gromady*). Pole *id* jest kluczem podstawowym w tej tabeli.

Tabela *gromady* zawiera pola: identyfikator gromady (*id* – klucz podstawowy) oraz nazwę gromady (*gromada*).

Poniżej pokazano przykładowe dane z obu tabel.

Tabela *zwierzeta*

id	gatunek	sposob_odzywiania	id_gromady
1	lew afrykanski	miesozerne	2
2	zyrafa	roslinozerne	2
3	krokodyl nilowy	miesozerne	3

Tabela *gromady*

id	gromada
1	ptaki
2	ssaki
3	gady
4	ryby

Zadanie 3.3. (0–1)

Dla tabel opisanych wyżej (i podanych danych przykładowych) listę nazw gatunków wszystkich ssaków z tabeli *zwierzeta* otrzymamy w wyniku zapytania

1.	SELECT gatunek FROM zwierzeta WHERE id_gromady = 2;	P	F
2.	SELECT gatunek FROM zwierzeta, gromady WHERE zwierzeta.id_gromady=gromady.id AND gromada = "ssaki";	P	F
3.	SELECT gatunek FROM zwierzeta INNER JOIN gromady ON zwierzeta.id_gromady=gromady.id WHERE gromada = "ssaki";	P	F
4.	SELECT gatunek FROM zwierzeta LEFT JOIN gromady ON zwierzeta.id_gromady=gromady.id WHERE gromada = "2";	P	F

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

Zadanie 3.4. (0–1)

Dla tabel opisanych wcześniej i podanych danych przykładowych

1.	<p>w wyniku zapytania:</p> <pre>SELECT gromada, count(*) FROM gromady INNER JOIN zwierzeta ON zwierzeta.id_gromady = gromady.id GROUP BY gromada;</pre> <p>otrzymamy nazwy gromad oraz liczby gatunków zwierząt należących do tych gromad.</p>	P	F
2.	<p>w wyniku zapytania:</p> <pre>SELECT gromada, count(*) FROM gromady INNER JOIN zwierzeta ON zwierzeta.id_gromady = gromady.id GROUP BY gromada;</pre> <p>otrzymamy liczbę różnych gromad z tabeli <i>gromady</i>.</p>	P	F
3.	<p>w wyniku zapytania:</p> <pre>SELECT count(*) FROM zwierzeta INNER JOIN gromady ON zwierzeta.id_gromady = gromady.id WHERE gromady.id=2;</pre> <p>otrzymamy liczbę gatunków ssaków z tabeli <i>zwierzeta</i>.</p>	P	F
4.	<p>w wyniku zapytania:</p> <pre>SELECT count(*) FROM zwierzeta, gromady WHERE zwierzeta.id_gromady = gromady.id AND gromady.gromada = "ssaki";</pre> <p>otrzymamy liczbę gatunków ssaków z tabeli <i>zwierzeta</i>.</p>	P	F

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

BRUDNOPIS (*nie podlega ocenie*)

Więcej arkuszy znajdziesz na stronie: arkusze.pl

INFORMATYKA

Poziom rozszerzony

Formuła 2015

INFORMATYKA

Poziom rozszerzony

Formuła 2015

INFORMATYKA

Poziom rozszerzony

Formuła 2015