

WYPEŁNIA ZDAJĄCY 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

	WYPEŁNIA ZDAJĄCY	Symbol arkusza
WYBRANE:		EINP -R1- 100 -2505
	(system operacyjny)	
	(program użytkowy)	
	(środowisko programistyczne)	

DATA: **14 maja 2025 г.**

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 10 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. Funkcja rekurencyjna

Dana jest rekurencyjna funkcja *przestaw*, której parametrem jest nieujemna liczba całkowita:

```
przestaw(n):
```

```
r \leftarrow n \mod 100

a \leftarrow r \operatorname{div} 10

b \leftarrow r \mod 10

n \leftarrow n \operatorname{div} 100

\mathbf{je\dot{z}eli} \ n > 0

w \leftarrow a + 10 * b + 100 * przestaw(n)

\mathbf{w} przeciwnym razie

\mathbf{je\dot{z}eli} \ a > 0

w \leftarrow a + 10 * b

\mathbf{w} przeciwnym razie

w \leftarrow b

wynikiem jest w
```

Uwaga:

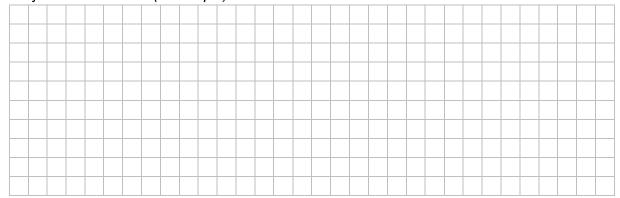
Operator mod oznacza resztę z dzielenia, natomiast div – część całkowitą z dzielenia.

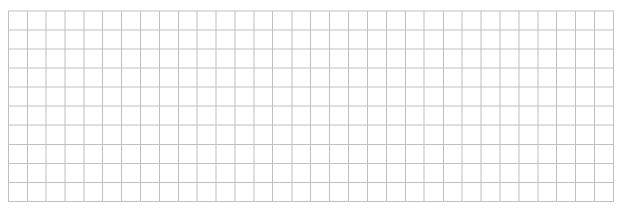
Zadanie 1.1. (0-3)

Uzupełnij tabelę – wpisz w drugiej kolumnie wynik funkcji *przestaw*(*n*) dla podanych wartości argumentu *n* oraz wpisz w trzeciej kolumnie liczbę wywołań funkcji *przestaw* łącznie z pierwszym wywołaniem z parametrem *n*.

n	Wynik działania funkcji przestaw	Liczba wywołań funkcji <i>przestaw</i>
316498	134689	3
43657688		
154005710		
998877665544321		

Miejsce na obliczenia (brudnopis)





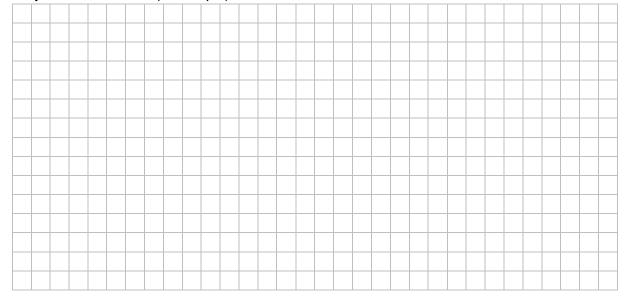
Zadanie 1.2. (0-2)

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

Niech n będzie liczbą k-cyfrową, gdzie k > 0. Liczba wywołań funkcji przestaw w zależności od k jest równa:

1.	$\frac{k}{2}$	Р	F
2.	$(k+1) \operatorname{div} 2$ (gdzie div oznacza dzielenie całkowite)	Р	F
3.	$\begin{cases} \frac{k}{2} & \text{gdy } k \text{ jest liczbą parzystą} \\ \frac{k+1}{2} & \text{gdy } k \text{ jest liczbą nieparzystą} \end{cases}$	Р	F
4.	$\frac{k+1}{2}$	Р	F

Miejsce na obliczenia (brudnopis)



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

Zadanie 2. Liczba falista

Liczba falista to liczba całkowita większa od 10, w której zapisie dziesiętnym występują na przemian tylko dwie różne cyfry.

Przykład: liczbami falistymi są 41414, 4545.

Niech n > 10 będzie liczbą całkowitą, w której zapisie dziesiętnym dwie ostatnie cyfry (najmniej znaczące) są różne od zera oraz różne od siebie. Liczbę n nazywamy liczbą bazowa, a jej ostatnie cyfry (z zachowaniem ich kolejności) nazywamy baza.

Z każdej liczby bazowej można otrzymać liczbę *falistą* o takiej samej *bazie* co liczba *n* i o takiej samej długości zapisu dziesiętnego.

Przykład:

Z liczby bazowej 78234 uzyskujemy liczbę falistą 43434

Z liczby bazowej 52786 – falistą 68686.

Zadanie 2.1. (0-2)

Uzupełnij tabelę – dla podanych wartości liczby bazowej podaj liczbę *falistą* o tej samej długości zapisu co liczba bazowa i o tej samej bazie.

Liczba bazowa	Liczba falista
326	
414141	
7732	
21289	

Zadanie 2.2. (0-4)

W postaci pseudokodu lub w wybranym języku programowania **napisz algorytm**, który dla liczby bazowej n obliczy liczbę falistą f o tej samej bazie co liczba n i o takiej samej długości zapisu dziesiętnego.

Przvkład:

Dla liczby 556621 wynikiem jest 212121, dla liczby 45621 wynikiem jest 12121.

Uwaga: Twój algorytm może używać wyłącznie zmiennych przechowujących pojedyncze liczby całkowite oraz może operować wyłącznie na liczbach całkowitych. W zapisie algorytmu możesz korzystać tylko z instrukcji sterujących, z operatorów arytmetycznych: dodawania, odejmowania, mnożenia, dzielenia, dzielenia całkowitego i reszty z dzielenia; z operatorów logicznych, z porównań i instrukcji przypisywania lub z samodzielnie napisanych funkcji i procedur wykorzystujących powyższe operacje. Zabronione jest używanie funkcji wbudowanych, dostępnych w językach programowania. Nie wolno w szczególności korzystać z żadnych funkcji zamiany z typu znakowego lub napisowego na liczbowy i odwrotnie.

Specyfikacja:

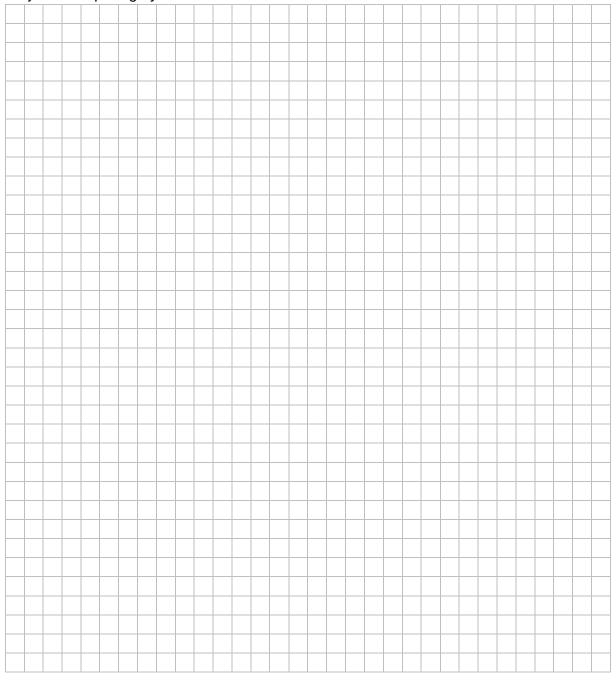
Dane:

n – liczba całkowita większa od 10, o różnych od zera i od siebie cyfrach dziesiątek i jedności.

Wynik:

f – liczba falista o tej samej bazie co liczba n i o takiej samej długości zapisu dziesiętnego

Miejsce na zapis algorytmu



	Nr zadania	2.1.	2.2.
Wypełnia	Maks. liczba pkt	2	4
egzaminator	Uzyskana liczba pkt		

Zadanie 3. Test

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

W każdym zadaniu punkt uzyskasz tylko za komplet poprawnych odpowiedzi.

Zadanie 3.1. (0-1)

Program typu keylogger służy do

1.	szyfrowania informacji do postaci uniemożliwiającej jej odczytanie bez zdefiniowanego klucza.	Р	F
2.	przechowywania danych logowania, w tym – haseł, w bezpiecznym miejscu na dysku użytkownika.	P	F
3.	generowania kodu, który umożliwia użytkownikowi bankowości elektronicznej wykonanie operacji.	Р	F
4.	przechwytywania i gromadzenia informacji o naciśniętych klawiszach.	Р	F

Zadanie 3.2. (0-1)

Liczba 1111 1111 1111 1111 $_2$ (zapisana w systemie binarym) jest równa

1.	333333334	Р	F
2.	7777778	P	F
3.	65535 ₁₀	Р	F
4.	$FFFF_{16}$	Р	F

Informacja do zadań 3.3. i 3.4.

Dane są tabele *Wlasciciele* i *Zwierzaki* połączone relacją jeden do wielu, kluczem Id_Własciciela:

Id_Wlasciciela	Imie	Nazwisko
1	Anna	Kowalska
2	Jan	Wierzbicki
3	Ewa	Nowak

Id_Zwierzaka	Imie	Gatunek	Id_Wlasciciela
1	As	Pies	1
2	Reksio	Pies	2
3	Kapsel	Pies	3
4	Mruczek	Kot	1
5	Kiciak	Kot	1
6	Puszek	Królik	3
7	Burek	Pies	2

Zadanie 3.3. (0-1)

Wynikiem zapytania:

SELECT Wlasciciele.Imie, Wlasciciele.Nazwisko, count(*)

FROM Wlasciciele

INNER JOIN Zwierzaki **ON** Wlasciciele.ld_Wlasciciela = Zwierzaki.ld_Wlasciciela **GROUP BY** Wlasciciele.lmie, Wlasciciele.Nazwisko;

jest

	Pies 4		
1.	Kot 2	Р	F
	Królik 1		
	Anna Kowalska 3		
2.	Ewa Nowak 2	Р	F
	Jan Wierzbicki 2		
	Anna Kowalska 10		
3.	Ewa Nowak 9	Р	F
	Jan Wierzbicki 9		
4.	lista imion i nazwisk właścicieli oraz liczb ich zwierząt	Р	F

Zadanie 3.4. (0-1)

Wynikiem zapytania:

SELECT Własciciele.Imie, Własciciele.Nazwisko, Zwierzaki.Imie

FROM Wlasciciele

INNER JOIN Zwierzaki ON Własciciele.ld_Własciciela = Zwierzaki.ld_Własciciela

WHERE Zwierzaki.lmie > "P"

ORDER BY Wlasciciele. Nazwisko ASC;

jest

1.	Ewa Nowak Puszek Jan Wierzbicki Reksio	Р	F
2.	pusty wynik	Р	F
3.	lista imion i nazwisk właścicieli zwierząt i imiona ich zwierząt zaczynające się literą P	Р	F
4.	Anna Kowalska Reksio	Р	F

Wypełnia egzaminator	Nr zadania	3.1.	3.2.	3.3.	3.4.
	Maks. liczba pkt	1	1	1	1
	Uzyskana liczba pkt				

BRUDNOPIS (nie podlega ocenie)

INFORMATYKA Poziom rozszerzony

Formula 2015

INFORMATYKA Poziom rozszerzony

Formula 2015

INFORMATYKA Poziom rozszerzony

Formula 2015