CENTRALNA KOMISJA EGZAMINACYJNA

Arkusz zawiera informacje prawnie chronione do momentu rozpoczęcia egzaminu.

WYPEŁNIA ZDAJĄCY

KOD

PESEL

Miejsce na naklejkę.
Sprawdź, czy kod na naklejce to **M-100**.
Jeżeli tak – przyklej naklejkę.
Jeżeli nie – zgłoś to nauczycielowi.

Egzamin maturalny *Formuła 2023*

INFORMATYKA

Poziom rozszerzony

Test diagnostyczny

Symbol arkusza
MINP-R0-**100**-2412

WYPEŁNIA ZDAJĄCY

WYBRANE:

.....
(system operacyjny)

.....
(program użytkowy)

.....
(środowisko programistyczne)

DATA: **10 grudnia 2024 r.**

GODZINA ROZPOCZĘCIA: **14:00**

CZAS TRWANIA: **210 minut**

LICZBA PUNKTÓW DO UZYSKANIA: **50**

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.

Więcej materiałów na lern.one

Układ graficzny
© CKE 2022

Instrukcja dla zdającego

1. Sprawdź, czy arkusz egzaminacyjny zawiera 16 stron (zadania 1–8) i czy dołączony jest do niego nośnik danych – podpisany DANE. Ewentualny brak zgłoś przewodniczącemu zespołu nadzorującego egzamin.
2. Na pierwszej stronie arkusza oraz na karcie odpowiedzi wpisz swój numer PESEL i przyklej naklejkę z kodem.
3. Na pierwszej stronie arkusza wpisz zadeklarowane (wybrane) przez Ciebie na egzamin: system operacyjny, program użytkowy oraz środowisko programistyczne.
4. Symbol zamieszczony w nagłówku zadania zwraca uwagę na to, że zadanie nie wymaga użycia komputera i odpowiedź do zadania należy zapisać tylko w miejscu na to przeznaczonym w arkuszu egzaminacyjnym.
5. Jeśli rozwiązaniem zadania lub jego części jest program komputerowy, to zapisz go w zadeklarowanym (wybranym) języku programowania i umieść w katalogu (folderze) oznaczonym Twoim numerem PESEL wszystkie utworzone przez siebie pliki w wersji źródłowej.
6. Jeśli rozwiązaniem zadania lub jego części jest baza danych utworzona z wykorzystaniem MySQL lub MariaDB, to umieść w katalogu (folderze) oznaczonym Twoim numerem PESEL treści zapytań w języku SQL oraz (przed zakończeniem egzaminu) wyeksportowaną całą bazę danych w formacie *.sql.
7. Pliki oddawane do oceny nazwij dokładnie tak, jak polecono w treści zadań, lub zapisz je pod nazwami (wraz z rozszerzeniem zgodnym z zadeklarowanym oprogramowaniem), jakie podajesz w arkuszu egzaminacyjnym. **Pliki o innych nazwach nie będą sprawdzane przez egzaminatora.**
   Pamiętaj, że zadania praktyczne niezawierające komputerowej realizacji rozwiązań zostaną ocenione na 0 punktów.
8. **Przed upływem czasu przeznaczonego na egzamin** zapisz w katalogu (folderze) oznaczonym Twoim numerem PESEL ostateczną wersję plików stanowiących rozwiązania zadań.
9. Pisz czytelnie. Używaj długopisu/pióra tylko z czarnym tuszem/atramentem.
10. Nie używaj korektora, a błędne zapisy wyraźnie przekreśl.
11. Nie wpisuj żadnych znaków w tabelkach przeznaczonych dla egzaminatora. Tabelki są umieszczone na marginesie przy każdym zadaniu.
12. Pamiętaj, że zapisy w brudnopisie nie będą oceniane.

Więcej materiałów na lern.one

Więcej materiałów na lern.one

Zadania egzaminacyjne są wydrukowane na następnych stronach.

Zadanie 1.

Dla danej dodatniej liczby całkowitej n niech $J(n)$ będzie ciągiem kolejnych pozycji jedynek (cyfr 1) w jej zapisie dwójkowym, zaczynając od pozycji najmniej znaczącej (ostatniej).

Pozycje cyfr w zapisie liczby numerujemy od 1 (zaczynamy od ostatniej pozycji).

Przykład 1.
Dla $n = 19$: $J(19) = (1, 2, 5)$, ponieważ $19_{10} = 10011_2$

1.1.
-1–2

Zadanie 1.1. (0–2)

Uzupełnij poniższą tabelę.

Liczba n	Zapis dwójkowy n	$J(n)$
19	10011	(1, 2, 5)
6	110	
42		
		(1, 2, 4, 7)

Miejsce na obliczenia (brudnopis)

Więcej materiałów na lern.one

Zadanie 1.2. (0–3)

1.2.
0–1–2–3

W pseudojęzyku lub w wybranym języku programowania zapisz algorytm, który dla danej liczby całkowitej dodatniej *n* wypisuje kolejne elementy ciągu *J*(*n*).

Uwaga: Twój algorytm może używać **wyłącznie zmiennych przechowujących liczby całkowite** oraz może operować **wyłącznie na liczbach całkowitych**. W zapisie możesz wykorzystać tylko operacje arytmetyczne: dodawanie, odejmowanie, mnożenie, dzielenie, dzielenie całkowite, resztę z dzielenia oraz porównywanie liczb, instrukcje sterujące, przypisania do zmiennych lub samodzielnie napisane funkcje, wykorzystujące wyżej wymienione operacje. **Zabronione** jest używanie funkcji wbudowanych oraz operatorów innych niż wymienione, nie wolno używać zwłaszcza żadnych funkcji zamiany typów danych.

Specyfikacja:

Dane:

n – dodatnia liczba całkowita

Wynik:

kolejne elementy ciągu *J*(*n*)

Przykład 2.

Dla *n* = 19 wynikiem algorytmu powinny być liczby: 1, 2, 5.

Algorytm:

Więcej materiałów na lern.one

Zadanie 2. Rekurencja

Przeanalizuj poniższą funkcję, której parametrami są nieujemna liczba całkowita x oraz dodatnia liczba całkowita $1 < p < 10$.

```
F(x, p) :
    jeżeli x = 0
        podaj wynik 0
    w przeciwnym przypadku
        c ← x mod p
        jeżeli c mod 2 = 1
            podaj wynik F( x div p, p ) + c
        w przeciwnym przypadku
            podaj wynik F( x div p, p ) – c
```

gdzie:
a mod b oznacza resztę z dzielenia liczby *a* przez liczbę *b*
a div b oznacza dzielenie całkowite liczby *a* przez liczbę *b*

2.1.
–1–
2–3

Zadanie 2.1. (0–3)

Uzupełnij poniższą tabelę. Dla podanych w tabeli wartości liczb x i p wpisz wynik funkcji F(x, p) oraz liczbę wywołań funkcji F, łącznie z pierwszym wywołaniem F(x, p).

x	***p***	**F(*x*, *p*)**	**Liczba wywołań funkcji F**
125	2	6	8
130	3		
220	4		

Miejsce na obliczenia (brudnopis)

Więcej materiałów na lern.one

Zadanie 2.2. (0–2)

2.2.
0–1–2

Uzupełnij poniższą tabelę. Dla podanych wartości p zapisz największą taką liczbę $x < 100$, dla której wynikiem funkcji $F(x, p)$ jest 0.

p	***x***
3	
4	

Miejsce na obliczenia (brudnopis)

Więcej materiałów na lern.one

Zadanie 3. Liczby czterocyfrowe

W pliku `liczby.txt` jest danych 2000 liczb całkowitych z zakresu [1000, 9999], każda w oddzielnym wierszu. Wszystkie liczby są różne.

Napisz program (lub kilka programów), który znajdzie odpowiedzi do poniższych zadań. Każdą odpowiedź zapisz w pliku `wyniki3.txt` i poprzedź ją numerem oznaczającym zadanie.

Do Twojej dyspozycji jest plik `liczby_przyklad.txt`, który zawiera 20 liczb w podobnej postaci. Odpowiedzi dla pliku `liczby_przyklad.txt` są podane w poleceniach zadań.

3.1.
–1–2

Zadanie 3.1. (0–2)

Znajdź w pliku `liczby.txt` takie liczby, które są kwadratami liczb całkowitych. Podaj, ile jest takich liczb, oraz podaj tę, która występuje w pliku jako pierwsza.

W pliku `liczby_przyklad.txt` są 2 kwadraty, pierwszy to 9801.

3.2.
–1–
2–3

Zadanie 3.2. (0–3)

Wypisz wszystkie liczby z pliku `liczby.txt`, które mają co najmniej 5 **różnych** dzielników pierwszych.

W pliku `liczby_przyklad.txt` jest jedna taka liczba: 2730.

3.3.
–1–
2–3

Zadanie 3.3. (0–3)

Do każdej liczby w pliku stosujemy następującą procedurę: z jej czterech cyfr tworzymy największą możliwą oraz najmniejszą możliwą liczbę, po czym drugą z nich odejmujemy od pierwszej. Przykładowo: dla liczby 3151 największa możliwa liczba to 5311, a najmniejsza to 1135, a więc wynik – ich różnica – wynosi 4176. Jeśli w liczbie występują zera, traktujemy je tak samo jak inne cyfry (na przykład dla liczby 2047 różnica to 7420 − 0247 = 7173).

Dla każdej liczby rozstrzygnij, czy różnica otrzymana w wyniku zastosowania podanej procedury jest mniejsza, większa czy równa danej liczbie. Podaj, ile razy otrzymamy różnicę mniejszą, ile razy – większą, a ile razy – równą danej liczbie. Ponadto wypisz każdą liczbę, dla której otrzymuje się w wyniku zastosowania podanej procedury różnicę równą danej liczbie.

Dla pliku `liczby_przyklad.txt` otrzymamy 8 razy mniejszą różnicę i 12 razy większą różnicę. Ani razu różnica nie będzie równa danej liczbie.

Do oceny oddajesz:

- plik `wyniki3.txt` – zawierający odpowiedzi do zadań 3.1.–3.3.
- plik(-i) zawierający(-e) kod(-y) źródłowy(-e) Twojego programu o nazwie(-ach) (uwaga: brak tego(tych) pliku(-ów) jest równoznaczny z brakiem rozwiązania zadania):

  Zadanie 3.1. ……

  Zadanie 3.2. ……

  Zadanie 3.3. ……

Więcej materiałów na lern.one

Zadanie 4. Prostokąty

W pliku `prostokaty.txt` opisano 5000 prostokątów, po jednym w każdym wierszu. Opis każdego prostokąta składa się z dwóch dodatnich liczb całkowitych nie większych od 40 000, oddzielonych spacją. Pierwszą z tych liczb będziemy nazywać wysokością prostokąta, a drugą – jego szerokością.

Przykład 1.
prostokąty:

o wysokości $h = 3$ i szerokości $s = 2$

o wysokości $h = 2$ i szerokości $s = 3$

Napisz program (lub kilka programów), które znajdą odpowiedzi do poniższych zadań. Każdą odpowiedź zapisz w pliku `wyniki4.txt` i poprzedź ją numerem oznaczającym zadanie.

Do Twojej dyspozycji jest plik `prostokaty_przyklad.txt`, który zawiera opis 20 prostokątów w tej samej postaci. Odpowiedzi dla pliku `prostokaty_przyklad.txt` są zawarte w poleceniach zadań.

Zadanie 4.1. (0–2)

4.1.
0–1–2

Podaj najmniejsze i największe pole powierzchni dla prostokątów z pliku z danymi `prostokaty.txt`.

Dla pliku `prostokaty_przyklad.txt` najmniejszym polem jest 252, a największym jest 9409.

Zadanie 4.2. (0–2)

4.2.
0–1–2

Powiemy, że prostokąt o szerokości s i wysokości h mieści się w prostokącie o szerokości s' i wysokości h' wtedy i tylko wtedy, gdy $s \le s'$ oraz $h \le h'$.

Znajdź najdłuższy ciąg **kolejnych** prostokątów w pliku `prostokaty.txt` taki, że każdy z tych prostokątów, poza pierwszym, mieści się w poprzednim. Jako odpowiedź podaj długość ciągu oraz wysokość i szerokość ostatniego prostokąta w tym ciągu.

Dla pliku `prostokaty_przyklad.txt` wynikiem jest 4 83 5 (ciąg 4-elementowy, ostatnim elementem ciągu jest prostokąt o wysokości 83 i szerokości 5).

Zadanie 4.3. (0–4)

4.3.
0–1–2–3–4

Prostokąty o tej samej wysokości można sklejać, żeby uzyskać prostokąt o takiej samej wysokości, co wysokość sklejanych prostokątów, oraz o szerokości, która jest sumą szerokości sklejanych prostokątów.

Podaj największe szerokości prostokątów, które można uzyskać przez sklejanie dokładnie 2, 3 i 5 prostokątów opisanych – odpowiednio – w 2, 3 i 5 różnych (niekoniecznie kolejnych) wierszach pliku `prostokaty.txt`.

Więcej materiałów na lern.one

Dla pliku `prostokaty_przyklad.txt` wyniki są następujące:
2 prostokąty: 183
3 prostokąty: 266
5 prostokątów: 409.

Do oceny oddajesz:

- plik `wyniki4.txt` – zawierający odpowiedzi do zadań 4.1.–4.3.
- plik(-i) zawierający(-e) kod(-y) źródłowy(-e) Twojego programu o nazwie(-ach) (uwaga: brak tego(tych) pliku(-ów) jest równoznaczny z brakiem rozwiązania zadania):

Zadanie 4.1. ……

Zadanie 4.2. ……

Zadanie 4.3. ……

5.
0–1

Zadanie 5. (0–1)

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

1.	Podpis elektroniczny jest gwarancją poufności treści (szyfruje zawartość dokumentu).	**P**	**F**
2.	Podpis elektroniczny służy do uwierzytelniania tożsamości podmiotu wystawiającego dokument.	**P**	**F**
3.	Podpis elektroniczny zabezpiecza przed utratą danych.	**P**	**F**

6.
–1–2

Zadanie 6. (0–2)

Wykonaj działania na liczbach zapisanych w systemie piątkowym. Wyniki podaj także w systemie piątkowym.

$2024_5 + 1044_5 =$ ……

$2024_5 - 1044_5 =$ ……

Miejsce na obliczenia (brudnopis)

Więcej materiałów na lern.one

Zadanie 7. Energia odnawialna

W kolejnych wierszach pliku `energia.txt` znajdują się dane o produkcji energii ze źródeł odnawialnych (wiatrowych i fotowoltaicznych). Każdy wiersz zawiera oddzielone znakami tabulacji informacje o dacie, o godzinie oraz o liczbie megawatogodzin (MWh) wyprodukowanych przez źródła wiatrowe, a także liczbie megawatogodzin wyprodukowanych przez źródła fotowoltaiczne w kwietniu 2024[1]. Dane są podane chronologicznie i nie zawierają luk (są podane dla każdego dnia i dla każdej godziny).

Fragment pliku:

Data	Godzina	Zrodla_wiatrowe	Zrodla_fotowoltaiczne
2024-04-01	1	3130,263	0,000
2024-04-01	2	2765,588	0,000
2024-04-01	3	2555,438	0,000
2024-04-01	4	2675,238	0,000
2024-04-01	5	2681,175	0,000
2024-04-01	6	2367,325	0,213
2024-04-01	7	2525,225	117,075
2024-04-01	8	2360,988	1162,075
2024-04-01	9	1940,838	2680,513

Z wykorzystaniem opisanych danych oraz dostępnych narzędzi informatycznych wykonaj następujące zadania. Wyniki zapisz w pliku tekstowym `wyniki7.txt`. Odpowiedź do każdego zadania poprzedź numerem zadania.

Zadanie 7.1. (0–2)

Podaj dzień, w którym łącznie wytworzono najwięcej MWh energii ze źródeł wiatrowych, oraz dzień, w którym łącznie wytworzono najwięcej MWh energii ze źródeł fotowoltaicznych w badanym okresie. W obu podanych kategoriach jest tylko jeden taki dzień.

7.1.
0–1–2

Zadanie 7.2. (0–3)

Podaj dla każdej godziny w dobie (od 1 do 24) średnią liczbę wyprodukowanych MWh energii ze źródeł fotowoltaicznych w tej godzinie w kwietniu. Podaj wyniki z dokładnością do dwóch miejsc po przecinku. Dla tego zestawienia wykonaj wykres kolumnowy. Wstaw tytuł wykresu i opisz osie.

7.2.
0–1–2–3

[1] Dane pochodzą ze strony https://www.pse.pl/dane-systemowe/funkcjonowanie-kse/raporty-dobowe-z-pracy-kse/generacja-zrodel-wiatrowych

Więcej materiałów na lern.one

7.3.
–1–
2–3

Zadanie 7.3. (0–3)

Znajdź najdłuższy przedział czasowy, w którym generowanie energii ze źródeł wiatrowych cały czas rosło. Podaj datę i godzinę początku przedziału oraz datę i godzinę końca przedziału. Dla podanych danych jest jeden najdłuższy przedział.

Przykład 1. Dla przykładowych danych:

```
2024-04-01  1   120
2024-04-01  2   111
2024-04-01  3   124
2024-04-01  4   125
2024-04-01  5   120
```

Datą i godziną początku przedziału jest 2024-04-01 godz. 2:00, a datą i godziną końca przedziału jest 2024-04-01 godz. 4:00.

7.4.
–1–
2–3

Zadanie 7.4. (0–3)

Zaprojektowano magazyny energii o łącznej pojemności 200 000 MWh. W magazynach jest przechowywana energia wygenerowana, ale niewykorzystana. Jeśli bieżące generowanie energii nie zaspokaja bieżącego zapotrzebowania, energia jest pobierana z magazynów. Rozliczenia odbywają się w jednostkach godzinowych.
Na potrzeby zadania wygenerowano dane – zapotrzebowanie na energię – i zapisano w pliku `zapotrzebowanie.txt`. Czasami jest ono mniejsze niż łączne generowanie energii, a czasami – większe. Zakładamy, że przesyłanie i przechowywanie energii nie przynosi strat energii.

Fragment pliku:

```
Data        Godzina     Zapotrzebowanie
2024-04-01  1           1524,564
2024-04-01  2           1307,093
2024-04-01  3           1222,506
2024-04-01  4           1225,012
2024-04-01  5           1836,463
```

Podaj, ile najmniej energii powinno być zmagazynowane w dniu 1 kwietnia o godzinie 00:00 (na początku dnia), aby kwietniowe zapotrzebowanie było zaspokojone, tzn. ani razu nie zabrakło energii. W obliczeniach weź pod uwagę generowanie energii ze źródeł odnawialnych i energię możliwą do pobrania z magazynów. Wynik podaj z dokładnością do 1 MWh.

Do oceny oddajesz:

- plik tekstowy `wyniki7.txt` – zawierający odpowiedzi do poszczególnych zadań (odpowiedź do każdego zadania powinna być poprzedzona jego numerem)

- plik zawierający wykres do zadania 7.2. o nazwie ……..
- plik(i) zawierający(e) komputerową realizację Twoich obliczeń o nazwie(-ach) (uwaga: brak tego(tych) pliku(-ów) jest równoznaczny z brakiem rozwiązania zadania):

  ……

Więcej materiałów na lern.one

Zadanie 8. Serwis samochodowy

W trzech plikach zgromadzono dane dotyczące usług świadczonych w pewnym serwisie samochodowym w listopadzie i w grudniu 2024 roku. Pierwszy wiersz w każdym z plików jest wierszem nagłówkowym i zawiera nazwy pól. Dane w wierszach są rozdzielone znakiem tabulacji.

Plik `samochody.txt` zawiera informacje o samochodach przyjętych do serwisu: *nr_rejestracyjny*, *marka*, *rok_produkcji*, *id_wlasciciela*.

Przykład:

```
nr_rejestracyjny marka   rok_produkcji  id_wlasciciela
LSW 3611         kia     2014           813
SY 7290          skoda   2016           739
```

Uwaga: W pliku zapisano informacje również o samochodach przyjętych do serwisu, ale jeszcze oczekujących na wykonanie usługi.

W pliku `uslugi.txt` każdy wiersz zawiera informacje o realizowanych usługach przez serwis samochodowy: *id_uslugi*, *usluga*, *rodzaj_uslugi*, *cena*.

Przykład:

```
id_uslugi  usluga                     rodzaj_uslugi   cena
8          wymiana filtra oleju       filtry          60
9          wymiana filtra kabinowego  filtry          60
```

W pliku `wykonane.txt` zapisano informacje o wykonanych usługach. Każdy wiersz zawiera następujące dane: *data*, *nr_rejestracyjny*, *id_uslugi*.

Przykład:

```
data           nr_rejestracyjny    id_uslugi
2024-11-04     SK 4178             2
2024-11-04     SD 1218             19
```

Z wykorzystaniem danych zawartych w podanych plikach oraz dostępnych narzędzi informatycznych podaj odpowiedzi do zadań 8.1.–8.3. Odpowiedzi zapisz w pliku `wyniki8.txt`, a każdą z nich poprzedź numerem odpowiedniego zadania.

Zadanie 8.1. (0–2)

8.1.
0–1–2

Podaj liczbę **różnych** samochodów spoza województwa śląskiego (nr rejestracyjny nie zaczyna się na literę S), które zostały przyjęte do serwisu i dla których wykonano co najmniej jedną usługę.

Więcej materiałów na lern.one

8.2.
-1–2

Zadanie 8.2. (0–2)

Podaj nazwę najliczniej wykonywanej usługi (*usługa*) w badanym okresie oraz podaj, ile razy ją wykonano. Wymień marki samochodów, dla których ta usługa **nie była** wykonywana.

8.3.
-1–2

Zadanie 8.3. (0–2)

Podaj identyfikator właściciela samochodu(ów), który zapłacił najwięcej za wszystkie wykonane usługi dla swojego/swoich samochodu(ów). Podaj numer(y) rejestracyjny(e) samochodu(ów) tego właściciela oraz nazwy wykonanych usług serwisowych.

8.4.
-1–2

Zadanie 8.4. (0–2)

Podaj kwotę przychodu, jaką serwis samochodowy uzyskał za wykonane usługi w listopadzie, i kwotę przychodu, jaką uzyskał w grudniu.

Do oceny oddajesz:

- plik tekstowy `wyniki8.txt` – zawierający odpowiedzi do poszczególnych zadań (odpowiedź do każdego zadania powinna być poprzedzona jego numerem)
- plik(i) zawierający(-e) komputerową realizację Twoich obliczeń o nazwie(-ach) (uwaga: brak tego(tych) pliku(-ów) jest równoznaczny z brakiem rozwiązania zadania):

.....

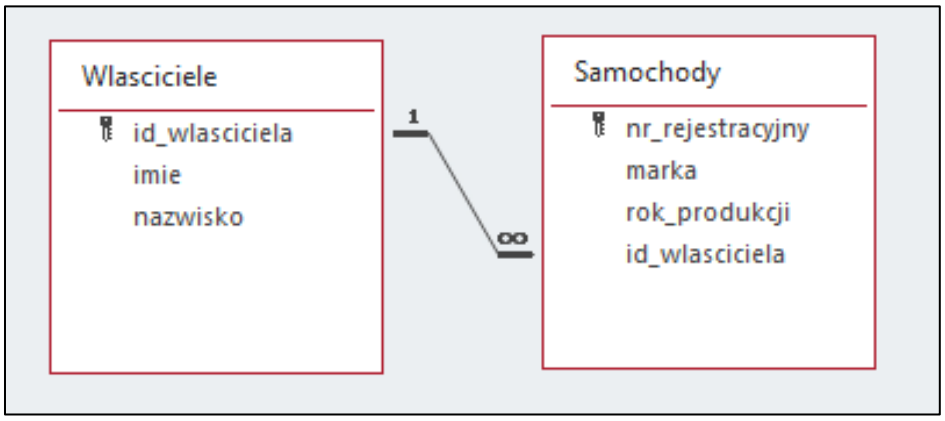
.....

Więcej materiałów na lern.one

Zadanie 8.5. (0–2)

8.5.
0–1–2

Do wcześniej opisanej bazy danych dołączono kolejną tabelę o nazwie `wlasciciele`, w której zapisano informacje o właścicielach serwisowanych samochodów: *id_wlasciciela*, *imie*, *nazwisko*.
Tabele `wlasciciele` i `samochody` są połączone relacją „jeden do wielu”:



Uwaga: jedna osoba może być właścicielem kilku samochodów. Każdy samochód ma tylko jednego właściciela.

Zapisz zapytanie SQL, w wyniku którego uzyskasz imiona i nazwiska właścicieli samochodów zarejestrowanych w Chorzowie (nr rejestracyjny rozpoczyna się od liter SH) i wyprodukowanych w 2020 roku. Nazwiska uporządkuj alfabetycznie.

Miejsce na zapis zapytania

Więcej materiałów na lern.one

BRUDNOPIS (*nie podlega ocenie*)

Więcej materiałów na lern.one

Więcej materiałów na lern.one

Więcej materiałów na lern.one

Więcej materiałów na lern.one

INFORMATYKA

Poziom rozszerzony

Formuła 2023

INFORMATYKA

Poziom rozszerzony

Formuła 2023

INFORMATYKA

Poziom rozszerzony

Formuła 2023

Więcej materiałów na lern.one