

Miejsce
na naklejkę
z kodem szkoły

dysleksja

☐

MIN-R1A1P-062

EGZAMIN MATURALNY Z INFORMATYKI

Arkusz I

POZIOM ROZSZERZONY

Czas pracy 90 minut

ARKUSZ I

MAJ
ROK 2006

Instrukcja dla zdającego

1. Sprawdź, czy arkusz egzaminacyjny zawiera 14 stron (zadania 1 – 4). 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. Wypełnij tę część karty odpowiedzi, którą koduje zdający. Nie wpisuj żadnych znaków w części przeznaczonej dla egzaminatora.
7. Na karcie odpowiedzi wpisz swoją datę urodzenia i PESEL. Zamaluj  pola odpowiadające cyfrom numeru PESEL. Błędne zaznaczenie otocz kółkiem  i zaznacz właściwe.

Życzymy powodzenia!

Za rozwiązanie
wszystkich zadań
można otrzymać
łącznie
40 punktów

Wypełnia zdający przed
rozpoczęciem pracy

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

PESEL ZDAJĄCEGO

--	--	--

KOD
ZDAJĄCEGO

Zadanie 1. Suma silni (11 pkt)

Pojęcie silni dla liczb naturalnych większych od zera definiuje się następująco:

$$n! = \begin{cases} 1 & \text{dla } n = 1 \\ (n-1)! * n & \text{dla } n > 1 \end{cases}$$

Rozpatrzmy funkcję $ss(n)$ zdefiniowaną następująco:

$$ss(n) = 1! + 2! + 3! + 4! + \dots + n! \quad (*)$$

gdzie n jest liczbą naturalną większą od zera.

- a) Podaj, ile mnożeń trzeba wykonać, aby obliczyć wartość funkcji $ss(n)$, korzystając wprost z podanych wzorów, tzn. obliczając każdą silnię we wzorze (*) oddzielnie. Uzupełnij poniższą tabelę.

Wartość funkcji	Liczba mnożeń
$ss(3)$	$0+1+2=3$
$ss(4)$	$0+1+2+3=6$
$ss(n)$	$0+1+2+3+\dots+n-1 = \frac{n*(n-1)}{2}$

- b) Zauważmy, że we wzorze na $ss(n)$, czynnik 2 występuje w $n-1$ silniach, czynnik 3 w $n-2$ silniach, ..., czynnik n w 1 silni. Korzystając z tej obserwacji przekształć wzór funkcji $ss(n)$ tak, aby można było policzyć wartość $ss(n)$, wykonując dokładnie $n-2$ mnożenia dla każdego $n \geq 2$. Uzupełnij poniższą tabelę (w ostatnim wierszu wypełnij tylko pusty prostokąt).

Wartość funkcji	Przekształcony wzór	Liczba mnożeń
$ss(1)$	1	0
$ss(2)$	1+2	0
$ss(3)$	$1+2*(1+3)$	1
$ss(4)$	$1+2*(1+3*(1+4))$	2
$ss(5)$	$1+2*(1+3*(1+4*(1+5)))$	3
$ss(n)$	$1+2*(1+3*(1+\dots(n-2)*(1+(n-1)*(1+n))\dots))$	$n-2$

Zapisz w wybranej przez siebie notacji (lista kroków, schemat blokowy lub język programowania) algorytm obliczania wartości funkcji $ss(n)$ zgodnie ze wzorem zapisanym przez Ciebie w tabeli. Podaj specyfikację dla tego algorytmu.

Dane: n – liczba naturalna, większa od 0

Wynik: $ss = 1! + 2! + 3! + 4! + \dots + n!$

Algorytm

Krok 1: Jeśli $n = 1$, to $ss := 1$ i idź do kroku 3,

w przeciwnym razie $ss := 1 + n$, $i := n - 1$

Krok 2: Dopóki $i > 1$ wykonuj $ss := 1 + i * ss$, $i := i - 1$

Krok 3: Zakończ wykonywanie algorytmu



Punktacja:

Części zadania	Maks.
a	2
b	9
Razem	11

Zadanie 2. Liczby pierwsze (13 pkt)

Poniżej przedstawiono algorytm wyznaczający wszystkie liczby pierwsze z przedziału $[2, N]$, wykorzystujący metodę Sita Eratostenesa. Po zakończeniu wykonywania tego algorytmu, dla każdego $i = 2, 3, \dots, N$, zachodzi $T[i]=0$, jeśli i jest liczbą pierwszą, natomiast $T[i]=1$, gdy i jest liczbą złożoną.

Dane: Liczba naturalna $N \geq 2$.

Wynik: Tablica $T[2..N]$, w której $T[i] = 0$, jeśli i jest liczbą pierwszą, natomiast $T[i]=1$, gdy i jest liczbą złożoną.

Krok 1. Dla $i = 2, 3, \dots, N$ wykonuj $T[i] := 0$

Krok 2. $i := 2$

Krok 3. Jeżeli $T[i] = 0$ to przejdź do kroku 4, w przeciwnym razie przejdź do kroku 6

Krok 4. $j := 2 * i$

Krok 5. Dopóki $j \leq N$ wykonuj

$T[j] := 1$

$j := j + i$

Krok 6. $i := i + 1$

Krok 7. Jeżeli $i < N$, to przejdź do kroku 3, w przeciwnym razie zakończ wykonywanie algorytmu

Uwaga: „:=” oznacza instrukcję przypisania.

- a) Dane są: liczba naturalna $M \geq 1$ i tablica $A[1..M]$ zawierająca M liczb naturalnych z przedziału $[2, N]$. Korzystając z powyższego algorytmu, zaprojektuj algorytm, wyznaczający te liczby z przedziału $[2, N]$, które nie są podzielne przez żadną z liczb $A[1], \dots, A[M]$. Zapisz go w wybranej przez siebie notacji (lista kroków, schemat blokowy lub język programowania) wraz ze specyfikacją.

Specyfikacja:

Dane: N, M – liczby naturalne, takie że $N > 1, M \geq 1$; tablica $A[1..M]$ liczb naturalnych z przedziału $[2, N]$.

Wynik: tablica $T[2..N]$ o wartościach 0 lub 1, w której $T[i]=0$

dla $i = 2, 3, \dots, N$ wtedy i tylko wtedy, gdy i nie jest podzielne przez żadną z liczb $A[1], \dots, A[M]$.

Krok 1. Dla $i = 2, 3, \dots, N$ wykonuj $T[i] := 0$

Krok 2. $k := 0$

Krok 3. Dopóki $k < M$ wykonuj

$k := k + 1$

$i := A[k]$

$j := i$

Dopóki $j \leq N$ wykonuj

$T[j] := 1$

$j := j + i$

Krok 4. Zakończ wykonywanie algorytmu

b) Do algorytmu opisanego na początku zadania wprowadzamy modyfikacje, po których ma on następującą postać:

- Krok 1. Dla $i = 2, 3, \dots, N$ wykonuj $T[i] := 0$
Krok 2. $i := 2$
Krok 3. Jeżeli $T[i] = 0$ to przejdź do kroku 4, w przeciwnym razie przejdź do kroku 6
Krok 4. $j := 2 * i$
Krok 5. Dopóki $j \leq N$ wykonuj
 $T[j] := T[j] + 1$
 $j := j + i$
Krok 6. $i := i + 1$
Krok 7. Jeżeli $i < N$, to przejdź do kroku 3, w przeciwnym razie zakończ wykonywanie algorytmu

Podaj, jakie będą wartości $T[13]$, $T[24]$, $T[33]$ po uruchomieniu tak zmodyfikowanego algorytmu dla $N=100$.

$$T[13] = 0$$

$$T[24] = 2$$

$$T[33] = 2$$

Podaj, dla jakiej wartości $T[i]$, dla i z przedziału $[2, N]$, i jest liczbą pierwszą.

i jest liczbą pierwszą, jeśli $T[i] = 0$.

Napisz, jaką własność liczb $i = 2, \dots, N$ określają wartości $T[i]$ po wykonaniu tak zmodyfikowanego algorytmu.

Wartość $T[i]$ oznacza liczbę dzielników właściwych liczby i , które są liczbami pierwszymi.

- c) Sito Eratostenesa służy do wyznaczania wszystkich liczb pierwszych z zadanego przedziału $[2, N]$. Podaj w wybranej przez siebie notacji (lista kroków, schemat blokowy lub język programowania) inny algorytm, który sprawdza, czy podana liczba naturalna $L > 1$ jest liczbą pierwszą. Zauważ, że chcemy sprawdzać pierwszość tylko liczby L , natomiast nie jest konieczne sprawdzanie pierwszości liczb mniejszych od L . Przy ocenie Twojego algorytmu będzie brana pod uwagę jego złożoność czasowa.

Specyfikacja:

Dane: Liczba naturalna $L > 1$.

Wynik: Komunikat „Tak”, jeśli L jest liczbą pierwszą, komunikat „Nie” w przeciwnym razie.

Krok 1: $j := 2$, $pierwsza := \text{true}$

Krok 2: Dopóki $(j \leq \sqrt{L})$ i $(pierwsza)$ wykonuj

$pierwsza := L \bmod j \neq 0$

$j := j + 1$

Krok 3: Jeśli $pierwsza$, to wypisz „Tak”, w przeciwnym razie wypisz „Nie”

Uwaga: $a \bmod b$ oznacza resztę z dzielenia liczby a przez liczbę b .



Punktacja:

Części zadania	Maks.
a	4
b	3
c	6
Razem	13

Zadanie 3. Baza danych (8 pkt)

Dyrektor szkoły dysponuje plikami **Uczniowie**, **Klasy** i **Przedmioty**.

Oto opisy wierszy w poszczególnych plikach:

Uczniowie – imię i nazwisko ucznia, numer jego legitymacji szkolnej oraz identyfikator klasy maturalnej, do której uczęszcza uczeń,

np.: *Jan Kowalski 7205 C*

Klasy – identyfikator klasy maturalnej i profil tej klasy,

np.: *C informatyczna*

Przedmioty – identyfikator przedmiotu, nazwa przedmiotu,

np.: *jp język polski*

Naszym celem jest zaprojektowanie bazy danych pozwalającej uzyskiwać informacje o tym, które przedmioty zostały wybrane na maturę przez poszczególnych uczniów.

W szczególności dyrektor chciałby uzyskiwać następujące informacje:

- wykaz uczniów, którzy zdają dany przedmiot (np. język angielski) na maturze,
- wykaz uczniów z klas informatycznych, którzy nie zdają matematyki na maturze.

Lista przedmiotów maturalnych (plik **Przedmioty**) może się zmieniać, dlatego nie należy przyjmować, że jest ona z góry ustalona. Zmiana listy przedmiotów maturalnych nie powinna wymagać zmiany struktury tabel bazy danych.

- a) W tabelach relacyjnej bazy danych istotne jest stosowanie kluczy. Podaj dwa przykłady zastosowania kluczy, zilustruj je na przykładzie poniższych tabel. Dla każdej z tych (przykładowych) tabel, wskaż kolumnę lub grupę kolumn, która jest jej kluczem podstawowym.

Uczniowie(Imię, Nazwisko, NumerLegitymacji, IdKlasy)

Klasy(IdKlasy, Profil)

Przedmioty(IdPrzedmiotu, Nazwa)

Przykłady zastosowania kluczy:

Klucz służy do identyfikowania wierszy w tabeli. Np. w tabeli **Uczniowie** NumerLegitymacji identyfikuje ucznia. Klucze służą do tworzenia związków między tabelami. Np. tabele **Uczniowie** i **Klasy** można połączyć za pomocą klucza IdKlasy.

W tabeli **Uczniowie**(Imię, Nazwisko, NumerLegitymacji, IdKlasy) kluczem jest NumerLegitymacji.

W tabeli **Klasy**(IdKlasy, Profil) kluczem jest IdKlasy.

W tabeli **Przedmioty**(IdPrzedmiotu, Nazwa) kluczem może być zarówno Nazwa, jak i IdPrzedmiotu.

- b) Zaprojektuj strukturę relacyjnej bazy danych, z której można uzyskać informacje potrzebne dyrektorowi. Przyjmij, że na maturze uczniowie mogą zdawać **dowolną** liczbę przedmiotów.
- i. Ustal, jakie tabele będą wchodziły w skład bazy danych (wykorzystaj definicje tabel z punktu a), jeśli to konieczne dodaj nowe tabele). Określ nazwy kolumn i typy danych dla kolumn tworzących poszczególne tabele w Twojej bazie danych. Przyjmij, że numer legitymacji jest liczbą naturalną z zakresu od 1 do 999999.

Uczniowie

Imię, Nazwisko : Tekst

NumerLegitymacji : Liczba

IdKlasy : Tekst

Klasy

IdKlasy, Profil : Tekst

Przedmioty

IdPrzedmiotu : Tekst

Nazwa : Tekst

Matura

NumerLegitymacji : Liczba

IdPrzedmiotu : Tekst

- ii. Zaprojektuj związki między tabelami Twojej bazy danych, właściwe dla struktury przechowywanej w bazie informacji. Określ rodzaj tych związków (jeden do jeden, jeden do wielu lub wiele do wielu).

Klasy : Uczniowie (poprzez pole IdKlasy) – typ jeden do wielu

Przedmioty : Matura (poprzez pole Nazwa: IdPrzedmiotu) – typ jeden do wielu

Uczniowie : Matura (poprzez pole NumerLegitymacji) – typ jeden do wielu

- c) Załóżmy, że pewna baza danych zawiera jedynie tabelę **Zgłoszenia** o kolumnach (Imię, Nazwisko, NumerLegitymacji, NazwaPrzedmiotu). Jeden wiersz takiej tabeli opisuje informację, iż uczeń o podanym imieniu, nazwisku i numerze legitymacji wybrał określony przedmiot do zdawania na maturze. Na przykładzie tej tabeli opisz zjawiska redundancji i anomalii modyfikacji (rozważ sytuację, gdy modyfikujemy numer legitymacji w jednym rekordzie). Uwzględnij fakt, że każdy uczeń może zdawać dowolną liczbę przedmiotów.

Zjawisko redundancji:

Przykładem redundancji jest przechowywanie dla każdego zgłoszenia, oprócz numeru legitymacji identyfikującej ucznia, także jego imienia i nazwiska.

Anomalia modyfikacji (przykład):

Zmiana nazwiska jednej osoby zdającej kilka przedmiotów wymaga wprowadzenia zmian we wszystkich wierszach dotyczących tej osoby. Pominięcie któregośkolwiek wiersza dotyczącego tej osoby, może spowodować utratę spójności danych.

Punktacja:

Części zadania	Maks.
a	2
b	4
c	2
Razem	8

Zadanie 4. Test (8 pkt)

Dla następujących zdań **zaznacz znakiem X** właściwe odpowiedzi.
(Uwaga: W każdym podpunkcie poprawna jest tylko jedna odpowiedź.)

- a) Adresy IP składają się z czterech liczb z zakresu od 0 do 255, które zapisuje się oddzielone kropkami, np. 130.11.121.94. Każda z tych liczb reprezentowana jest w komputerze na ośmiu bitach. Wśród adresów IP wyróżniamy m.in. adresy klasy B, w których pierwsza z liczb zapisana binarnie na ośmiu bitach, ma na dwóch pierwszych pozycjach (licząc od lewej strony) wartości odpowiednio 1 i 0. Który z poniższych adresów jest adresem IP typu B?
- ☒ 131.125.94.11
 - ☐ 141.125.294.111
 - ☐ 201.93.93.93
- b) Liczba 2101 oznacza
- ☐ 13 zapisane w systemie binarnym.
 - ☒ 64 zapisane w systemie trójkowym.
 - ☐ 1099 zapisane w systemie ósemkowym.
- c) Najmniejszą jednostką informacji jest
- ☒ bit.
 - ☐ bajt.
 - ☐ znak.
- d) System operacyjny to
- ☐ program umożliwiający szybką realizację operacji matematycznych.
 - ☒ zbiór programów zarządzających pracą komputera.
 - ☐ program służący wyłącznie do formatowania dysków i kopiowania plików.
- e) Do metod ochrony poufności danych należy
- ☐ systematyczne gromadzenie danych w pamięci operacyjnej.
 - ☒ zabezpieczenie dostępu do danych przez hasło.
 - ☐ stosowanie programów archiwizujących.
- f) Portal internetowy to
- ☐ program o funkcjach podobnych do programów Internet Explorer, Mozilla, Opera.
 - ☐ inna nazwa otoczenia sieciowego.
 - ☒ wielotematyczny serwis internetowy.

- g) Które z poniższych czynności są przykładami kodowania informacji?
- ☒ Zastąpienie znaków tworzących tekst innymi znakami w sposób pozwalający odtworzyć tekst oryginalny.
 - ☐ Usunięcie losowo wybranych liter z tekstu wiadomości.
 - ☐ Ukrywanie przekazywanych wiadomości poprzez dobór odpowiednich uprawnień i atrybutów.
- h) Grafika rastrowa to sposób tworzenia i przechowywania w komputerze obrazów, które są reprezentowane w postaci
- ☐ równań figur geometrycznych (odcinków, łuków, okręgów, elips).
 - ☒ siatki niezależnie traktowanych pikseli.
 - ☐ zbiorów odcinków.

Punktacja:

Zadanie	Maks.
Razem	8

BRUDNOPIS