

Arkusz zawiera informacje prawnie chronione do momentu rozpoczęcia egzaminu.

|  |  |
| --- | --- |
| **WYPEŁNIA ZESPÓŁ NADZORUJĄCY** | ***Miejsce na naklejkę.****Sprawdź, czy kod na naklejce to***M-660** |
|  |
|  **KOD PESEL** |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Układ graficzny

© CKE 2022

|  |
| --- |
| **EGZAMIN MATURALNY****INFORMATYKA****– POZIOM ROZSZERZONY****Arkusz pokazowy**Termin: **4 marca 2022 r.**Czas pracy: **do 315 minut** Liczba punktów do uzyskania: **50** |

|  |  |
| --- | --- |
| **WYPEŁNIA ZESPÓŁ NADZORUJĄCY** | WYBRANE:.................................................(system operacyjny).................................................(program użytkowy).................................................(język programowania i środowisko programistyczne) |

|  |
| --- |
| **Instrukcja dla zdającego**1. Sprawdź, czy arkusz egzaminacyjny zawiera 7 zadań i czy dołączony jest do niego nośnik danych – podpisany DANE\_PR. Ewentualny brak zgłoś przewodniczącemu zespołu nadzorującego egzamin.
2. Pamiętaj, że zapisy w brudnopisie nie będą oceniane.
3. Zapis [arkusz] zamieszczony w nagłówku zadania oznacza, że zadanie nie wymaga użycia komputera i odpowiedź do niego należy dołączyć do arkusza egzaminacyjnego.
4. Pliki oddawane do oceny nazwij dokładnie tak, jak polecono w treści zadań, lub zapisz je pod nazwami (wraz z rozszerzeniem zgodnymz zadeklarowanym oprogramowaniem), jakie podajesz w arkuszu egzaminacyjnym.Pliki o innych nazwach nie będą sprawdzane przez egzaminatora.

MINP-R0-**660**-2203 |

 Zadanie 1. Szachy
 Uwaga: do rozwiązania zadań 1.1.–1.3. nie jest potrzebna znajomość reguł gry w szachy.

W pliku szachy.txt znajduje się zapis partii szachów, jaką w 2020 roku rozegrali polski arcymistrz Jan Krzysztof Duda oraz mistrz świata Magnus Carlssen. Zapis partii składa się z opisów 125 plansz przedstawiających stany gry(położenie bierek na szachownicy) po kolejnych posunięciach każdego z graczy.

Opis każdej planszy składa się z: 8 wierszy tekstu po 8 znaków w każdym wierszu
kolejne znaki w wierszach oznaczają:
znak '.' − puste pole
wielkie litery −białe bierki (czyli białe figury i pionki)
małe litery − czarne bierki,
oznaczenia bierek to: K/k - król, H/h - hetman, W/w - wieża, G/g - goniec, S/s - skoczek,
P/p - pionek.

Przykład

Na rysunku przedstawiono planszę z przykładowym stanem gry.

w

s

g

h

k

g

s

w

p

p

p

p

p

p

p

p

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

.

P

.

.

.

.

.

.

.

.

.

.

.

P

P

P

P

.

P

P

P

G

H

K

G

S

W

S

W

Wygląd planszy na rysunku odpowiada stanie gry. Stan ten można go opisać w następujący sposób.

wsghkgsw

pppppppp

........

........

....P...

........

PPPP.PPP

WSGHKGSW

Dla zachowania czytelności, po każdym opisie następuje pojedynczy pusty wiersz. W dalszej części, zamiast „opis planszy”, będziemy pisać krótko „plansza”.

Napisz program(-y), który(-e) znajdzie(-dą) odpowiedzi do poniższych zadań. Do Twojej dyspozycji jest plik szachy\_przyklad.txt, który zawiera 9 plansz zapisanych w podanym wyżejformacie. Odpowiedzi dla pliku szachy\_przyklad.txt podano w treści poszczególnych zadań. Pamiętaj, że Twój(-e) program(-y) musi(-szą) działać dla 125 plansz.

 Zadanie 1.1. (0−3) Podaj, na ilu planszach znajduje się przynajmniej jedna pusta kolumna, czyli taka,na polach której nie stoi żadna bierka.Podaj także największą liczbę pustych kolumn na jednej z tych plansz.

Odpowiedź dla pliku szachy\_przyklad.txt:
7 5
(7 plansz z pustymi kolumnami, największa liczba pustych kolumn na planszy - 5).

Do oceny oddajesz:
- plik zadanie1\_1.txt zawierający odpowiedź do zadania (dwie liczby oddzielone spacją –liczba plansz z pustymi kolumnami oraz największa liczba pustych kolumn na planszy)
- plik(-i) z komputerową realizacją zadania (kodem programu) o nazwie (nazwach)

 Zadanie 1.2. (0−3)
 Rozstrzygnij, ile razy w trakcie gry (inaczej, na ilu planszach zapisanych w pliku szachy.txt) nastąpiła sytuacja, w której jest równowaga – jest tyle samo i takich samych czarnych bierek co białych. Podaj liczbę takich plansz, a także najmniejszą liczbę bierek (łącznie białych i czarnych) na planszy w stanie równowagi.

Przykład:
Plansza A jest w równowadze, a plansza B – nie jest w równowadze (czarne i białe nie mają takich samych bierek).

A:
.k......
........
........
....s...
....S...
........
........
.......K

B:
.p......
........
........
....s...
....S...
........
........
.......K

Odpowiedź dla pliku szachy\_przyklad.txt:
6 4
(6 plansz w równowadze, najmniejsza liczba bierek na planszy w stanie równowagi - 4)

Do oceny oddajesz:
- plik zadanie1\_2.txt zawierający odpowiedź do zadania (dwie liczby oddzielone spacją –liczba plansz w stanie równowagi oraz najmniejsza liczba bierek na planszy w stanie równowagi)
- plik(-i) z komputerową realizacją zadania (kodem programu) o nazwie (nazwach)

 Zadanie 1.3. (0−4)
 Wieża szachuje króla przeciwnego gracza, jeśli znajduje się w tym samym wierszu lub w tej samej kolumnie co król i pomiędzy nimi nie ma żadnej innej bierki.

Oblicz i podaj, na ilu planszach biała wieża szachuje czarnego króla oraz na ilu planszach czarna wieża szachuje białego króla.

Odpowiedź dla pliku szachy\_przyklad.txt:
2 0
(2 razy biała wieża szachuje czarnego króla, 0 razy czarna wieża szachuje białego króla).

Do oceny oddajesz:
- plik zadanie1\_3.txt zawierający odpowiedź do zadania (dwie liczby oddzielone spacją –liczba plansz, na których biała wieża szachuje czarnego króla, i liczba plansz, na których czarna wieża szachuje białego króla)
- plik(-i) z komputerową realizacją zadania (kodem programu) o nazwie (nazwach)
 Zadanie 2. Gra
 W tym zadaniu analizujemy prostą grę polegającą na ustawianiu pionków na jednowymiarowej planszy B składającej się z s+1 pól, ponumerowanych 0, 1, …, s, dla pewnej dodatniej liczby całkowitej s. B[i] oznacza i-te pole na planszy.

Na początku gry na planszy stawiamy tylko 1 pionek na polu o numerze 0. Gra składa się z n tur. Sposób stawiania pionków w turach jest zadany przez n dodatnich liczb całkowitych zapisanych w tablicy A[1..n]. W k-tej turze pionki stawiamy zgodnie z procedurą Tura(k) zdefiniowaną następująco:

Tura(k)
 dla i = s, s–1, ..., A[k] wykonuj
 jeśli na polu B[i - A[k]] znajduje się pionek i pole B[i] jest puste
 postaw pionek na polu B[i]

Formalnie rozgrywkę na planszy można zdefiniować teraz następująco:

Dane:

 s, n – dodatnie liczby całkowite

 A[1..n] – tablica n dodatnich liczb całkowitych

Wynik:

 B[0..s] – plansza do gry z ustawionymi pionkami, B[i] – i-te pole planszy

Rozgrywka:

 postaw pionek na B[0];

 dla k = 1, 2, …, n wykonuj

 Tura(k)

Gra kończy się sukcesem, gdy na polu B[s] stoi pionek.

 Zadanie 2.1.(0−2) [arkusz]

 Podaj wyniki gry (TAK - sukces lub NIE - porażka) dla podanych tablicy A i liczby s.
Po liczbie s zapisz TAK lub NIE.

A: [1, 2, 3 ]

s: 5

Sukces: TAK

A: [1, 2, 5, 10]

s: 14

Sukces: ….

A: [13, 5, 5, 2, 7]

s: 17

Sukces: ….

A: [7, 6, 5, 4, 3, 2, 1]

s: 25

Sukces: ….

 Zadanie 2.2.(0−2) [arkusz]
 Niech A = [5, 10, 15, 20, 25, ..., 95, 100] oraz s = 500. Na ilu polach planszy B znajdą się pionki po zakończeniu gry?

Odpowiedź: ….

 Zadanie 2.3.(0−2) [arkusz]

 Podaj przykładową zawartość co najwyżej 10 elementowej tablicy A, dla której dla każdego s = 1, 2, 3, …, 200 gra kończy się sukcesem.

Odpowiedź: ….

 Zadanie 3. Potęgowanie modulo
 Rozważmy operację potęgowania modularnego stosowaną na przykład w algorytmie RSA.

Liczbę a podnosimy do potęgi x, po czym bierzemy resztę z dzielenia otrzymanej liczby przez ustaloną liczbę M, dzięki czemu otrzymujemy wynik

b = axmodM,

gdzie:
a, M – dodatnie liczby całkowite,
x – nieujemna liczba całkowita.

Mówimy wtedy, że ax modulo M równa się b.

Przykład:

Dla a = 2, x = 5, M = 7 liczymy resztę z dzielenia 25 (czyli 32) przez 7, zatem b = 4.

Dla a = 3, x = 3 i M = 11 mamy b = 33mod 11 = 5,

natomiast dla a = 10, x = 2 i M = 13 wynikiem jest b = 102mod 13 = 9.

 Zadanie 3.1. (0−2) [arkusz]

 Podaj brakującą liczbę (x lub b), dla której axmod M = b.

Po każdej liczbie M zapisz brakującą liczbę x lub b.

M: 7, a = 2, x = 5, b = 4
M: 11, a = 3, x = 3, b = ….

M: 31, a = 5, x = …., b = 25
M: 59, a = 2, x = …., b = 5
M: 80, a = 9, x = 2, b = ….

 Zadanie 3.2. (0−4) [arkusz]
 Zapisz w wybranej przez siebie notacji (w postaci pseudokodu lub w wybranym języku programowania) algorytm, który gdy są dane liczby a, x i M, obliczy b = ax mod M. Aby otrzymać maksymalną liczbę punktów, Twój algorytm powinien wykonywać O(log x) operacji arytmetycznych wymienionych w poniższej uwadze.

Uwaga: W zapisie algorytmu możesz wykorzystać tylko operacje arytmetyczne: dodawanie, odejmowanie, mnożenie, dzielenie, dzielenie całkowite, resztę z dzielenia, oraz porównywanie liczb; instrukcje sterujące i przypisania do zmiennych lub samodzielnie napisane funkcje zawierające wyżej wymienione operacje.

Specyfikacja:

Dane:

 a  −liczba całkowita dodatnia

 x  −nieujemna liczba całkowita

 M −liczba całkowita dodatnia

Wynik:

 b −nieujemna liczba całkowita o wartości równej axmodM

Algorytm: ….

 Informacja do zadań 3.3.−3.5.
 W pliku liczby.txt jest 1000 wierszy, w każdym po trzy nieujemne liczby całkowite, kolejno M, a, b, oddzielone pojedynczymi spacjami. Liczby w pliku są nie większe niż 10 000, a ponadto wszystkie liczby M i a są większe bądź równe 2.

Napisz program(-y), który(-e) znajdzie(-dą) odpowiedzi na poniższe pytania. Każdą odpowiedź zapisz w pliku wyniki3.txt i poprzedź ją numerem odpowiedniego zadania.

Do Twojej dyspozycji jest plik liczby\_przyklad.txt, w którym zapisano 5 wierszy w formacie opisanym wyżej. Odpowiedzi dla pliku przykładowego są podane przy odpowiednich zadaniach - możesz ich użyć, aby sprawdzić poprawność działania swojego programu.

 Zadanie 3.3. (0−2)
 Oblicz, w ilu wierszach pliku liczby.txt liczba M jest liczbą pierwszą.

Dla pliku liczby\_przyklad.txt odpowiedź wynosi 2.

Zadanie 3.4. (0−2)

Oblicz, w ilu wierszach pliku liczby.txt pierwsze dwie zapisane liczby (M i a) są względnie pierwsze (to znaczy ich największym wspólnym dzielnikiem jest 1).

Dla pliku liczby\_przyklad.txt odpowiedź wynosi 3.

 Zadanie 3.5. (0−2)
 Dla każdej trójki liczb (M, a, b) zapisanej w jednym wierszu pliku rozstrzygnij, czy możliwe jest znalezienie takiego xz przedziału [0..M– 1], dla którego axmod M = b. Podaj, dla ilu trójek zachodzi taka sytuacja.

Dla pliku liczby\_przyklad.txt odpowiedź wynosi4.

Do oceny oddajesz:

plik tekstowy wyniki3.txt zawierający odpowiedzi do poszczególnych zadań (odpowiedź do każdego zadania powinna być poprzedzona jego numerem)

pliki zawierające kody źródłowe Twoich programów o nazwach odpowiednio:

zadanie 3.3. ….
zadanie 3.4. ….
zadanie 3.5. ….

 Zadanie 4. Brenna
 W pliku tekstowym brenna.txt w każdym wierszu zapisano daty i godziny oraz wyniki pomiarów temperatury (w stopniach C) i opadu (w cm) w stacji meteorologicznej Brenna z okresu od 01.01.2019 do 31.12.2019. Dane w wierszach pliku rozdzielone są znakiem tabulacji.

Przykładowy fragment pliku:

data temperatura opad

01.01.2019 00:00 -3,2 0

01.01.2019 01:00 -3 0

01.01.2019 02:00 -3,3 0

01.01.2019 03:00 -3,7 0

01.01.2019 04:00 -3,8 0

01.01.2019 05:00 -3,5 0

Z wykorzystaniem danych zawartych w pliku brenna.txt oraz dostępnych narzędzi informatycznych wykonaj poniższe polecenia. Każdą odpowiedź umieść w pliku wyniki4.txt i poprzedź oznaczeniem odpowiedniego zadania od 4.1.do 4.5.

 Zadanie 4.1. (0−2)
 Podaj dzień, w którym dobowa amplituda temperatury była najwyższa oraz wartość tej amplitudy.
Uwaga: Amplituda to różnica między temperaturą najwyższą, a najniższą.

 Zadanie 4.2. (0−3)
 Dokonaj analizy wszystkich danych i podaj, dla każdej godziny w dobie zegarowej, średnią temperaturę w całym roku. Wyniki zaokrąglij do dwóch miejsc po przecinku. Na podstawie otrzymanego zestawienia utwórz wykres liniowy.

 Zadanie 4.3. (0−2)

 Jeżeli temperatura jest większa od zera oraz opad jest większy od zera, to przyjmujemy, że pada deszcz. Podaj, ile godzin trwał najdłuższy ciąg pomiarów (nieprzerwany), gdy padał deszcz. Podaj datę i godzinę rozpoczęcia opadu, datę i godzinę zakończenia opadu oraz łączną sumę opadów w tym czasie.

 Zadanie 4.4. (0−3)

 Zakładamy, że jeżeli temperatura jest mniejsza lub równa zero i jest opad (czyli opad > 0), to oznacza, że pada śnieg.

Opady śniegu są monitorowane przez przedsiębiorstwo odśnieżania. Zliczana jest łączna wysokość opadów śniegu z kolejnych godzin, w których nie padał deszcz. Jeżeli łączna wysokość opadów śniegu nieprzerwanych deszczem przekroczy 4 cm, to w następnej godzinie na drogi wyjeżdżają pługi odśnieżające.

Przykład:

01.01.2019 09:00 -2,5 0

01.01.2019 10:00 -2,2 0,1

01.01.2019 11:00 -1,8 0

01.01.2019 12:00 -1,2 0,2

01.01.2019 13:00 -1,1 0,2

01.01.2019 14:00 -0,7 0,6

01.01.2019 15:00 -0,5 0,6

01.01.2019 16:00 -0,2 0,8

01.01.2019 17:00 0,2 2,5

Dla podanych danych od godziny 9:00 do 16:00 łączna wysokość opadów śniegu to 2,5 cm, o godzinie 17:00 spadł deszcz. W tym przypadku pługi nie wyjechały, a zliczanie opadów śniegu w kolejnych godzinach zacznie się od 0.

Odśnieżanie trwa dokładnie 1 godzinę (opad śniegu, który nastąpił w czasie pracy pługów, zostaje usunięty z dróg w tym cyklu odśnieżania). Opad z kolejnej godziny po odśnieżaniu staje się częścią sumy opadów dla następnego cyklu odśnieżania.

Podaj, ile razy pługi odśnieżające wyjeżdżały na drogi Brennej.

Podaj dzień, w którym pługi wyjeżdżały najwięcej razy oraz liczbę tych wyjazdów.

Do oceny oddajesz:

- plik wyniki4.txt zawierający odpowiedzi do zadań 4.1.-4.4.

- plik zawierający wykres do zadania 4.2. o nazwie ….

- plik(-i) z komputerową realizacją zadania (kodem programu) o nazwie (nazwach)

 Zadanie 5. Statki

 W trzech plikach tekstowych o nazwach statki.txt, kody.txt, przybycia.txt, zapisano dane (w postaci tabeli) o statkach przybywających do portu w Szczecinie w okresie od 1.01.2016 do 31.12.2019. Pierwszy wiersz każdego z plików jest wierszem nagłówkowym, a dane w wierszach rozdzielono średnikami.

Plik o nazwie statki.txt zawiera informacje o 2 192 różnych statkach, które cumowały w szczecińskim porcie:

− Nr\_IMO – siedmiocyfrowy, unikatowy identyfikator statku

− Nazwa\_statku – nazwa statku (tekst do 50 znaków)

− Ladownosc – ładowność statku (maksymalnie 5-cyfrowa liczba).

Przykład:

Nr\_IMO;Nazwa\_statku;Ladownosc

5111696;RONJA;127

5255777;NORDSTJERNEN;2191

Plik o nazwie przybycia.txt zawiera zestawienie kolejnych statków przypływających do portu (10 034 wierszy) we wspomnianym okresie. W każdym wierszu znajduje się:

− LP – liczba porządkowa (liczba maksymalnie 5-cyfrowa)

−Data\_przybycia – data przybycia (w formacie dd.mm.rrrr)

−Nr\_IMO – siedmiocyfrowy identyfikator statku

−Bandera – symbol przynależności państwowej statku (tekst dwuznakowy)

−Nabrzeze – miejsce postoju statku (tekst do 20 znaków).

Uwaga: bandera odnotowywana jest w pliku przybycia.txt, ponieważ statek może zmienić banderę, np. jeżeli zakupi go inny armator.

Przykład:

LP;Data\_przybycia;Nr\_IMO;Bandera;Nabrzeze

1;01.01.2016;8415653;NO;HUTA (KRA)

2;01.01.2016;9201803;NL;CZESKIE

Plik o nazwie kody.txt zawiera przyporządkowanie dwuliterowego kodu bandery do państwa. W każdym z 240 wierszy znajduje się:

−Bandera – tekst dwuznakowy

−Nazwa\_kraju – nazwa kraju (tekst do 50 znaków)

−Kontynent – kontynent (tekst do 20 znaków).

Przykład:

Bandera;Nazwa\_kraju;Kontynent

AF;AFGANISTAN;AZJA

AL;ALBANIA;EUROPA

Z wykorzystaniem danych zawartych w podanych plikach oraz dostępnych narzędzi informatycznych wykonaj poniższe polecenia. Każdą odpowiedź umieść w pliku wyniki5.txt i poprzedź oznaczeniem odpowiedniego zadania od 5.1. do 5.3.

 Zadanie 5.1. (0−1)

 Podaj liczby wpłynięć statków do portu w Szczecinie w poszczególnych latach: 2016, 2017, 2018 i 2019.

 Zadanie 5.2. (0−2)

 Dla każdego nabrzeża podaj nazwę i ładowność największego statku, jaki przy nim cumował.

 Zadanie 5.3. (0−2)

 Podaj nazwy nabrzeży portu, przy których nie cumowały statki z Europy.

Do oceny oddajesz:

- plik wyniki5.txt zawierający odpowiedzi do zadań 5.1.-5.3.

- plik(-i) z komputerową realizacją zadania (kodem programu) o nazwie (nazwach)

Informacja do zadań 5.4. i 5.5.

Do wcześniej opisanych tabel dołączamy kolejną o nazwie Armator, w której zapisano informacje o armatorach eksploatujących statki. Tabela Armator zawiera następujące pola:

−Id\_armatora – unikatowy numer armatora

−Armator – nazwa armatora

−Siedziba\_armatora – miasto, w którym armator ma swoją siedzibę

−Typ\_dzialalnosci – dominujący typ działalności, np. przewóz ładunków,

eksploatacja morska, transport pasażerów itp.

Ponadto, do tabeli Przybycia dodano pole Id\_armatora.

Poniższy schemat przedstawia wszystkie tabele.

Przybycia

 LP
 Data\_przybycia

 Nr\_IMO
 Bandera
 Nabrzeze

 Id\_amator

Statki

 Nr\_IMO
 Nazwa\_statku

 Ladownosc

Kody

 Bandera
 Nazwa\_kraju

 Kontynent

Amator

 Id\_amator
 Amator

 Siedziba\_amatora
 Typ\_dzialalnosci

 Zadanie 5.4. (0−2) [arkusz]

 Napisz w języku SQL zapytanie, w wyniku którego uzyskasz zestawienie typów działalności i liczby armatorów, którzy taką właśnie działalność prowadzą.

Odpowiedź: ….

 Zadanie 5.5. (0−2) [arkusz]
 Napisz w języku SQL zapytanie, w wyniku którego uzyskasz zestawienie różnych nazw statków eksploatowanych przez armatora o nazwie XYZ, które cumowały w porcie.

Odpowiedź: …..

 Zadanie 6. Wikipedia (0−1) [arkusz]
 Oceń prawdziwość podanych zakończeń zdań. Po numerze zakończenia zdania zapisz P, jeśli zdanie jest prawdziwe, albo F – jeśli jest fałszywe.

W myśl polskiego prawa dozwolone jest

1. skopiowanie treści z wikipedia.org i użycie jej jako części własnego referatu, z nieznaczną zmianą tak, aby sformułowania nie były dokładnie takie same.
2. użycie na własnym blogu zdjęcia z wikipedia.org z uwagą "zdjęcie pochodzi z wikipedia.org" i identyfikatorem autora.
3. wklejenie własnego referatu jako część odpowiedniego hasła na wikipedia.org.

 Zadanie 7. (0−2) [arkusz]

 Przy transakcjach wykonywanych w pewnym sklepie internetowym potrzebne są między innymi trzy wrażliwe informacje: login (nazwa użytkownika), hasło do serwisu i dane karty kredytowej.

Dla każdej z tych informacji wskaż zalecany i prawidłowy (zgodny z powszechnie przyjętymi praktykami bezpieczeństwa) sposób postępowania z danymi.

Po numerze sposobu postępowania z danymi przy każdej danej zapisz P jeśli sposób postępowania jest zalecany i prawidłowy.

Sposób postępowania z danymi w bazie danych sklepu:
1. należy zapisać w bazie danych sklepu w całości

login : ….
hasło: ….
dane karty kredytowej: ….

2. nie należy przechowywać w bazie danych sklepu w żadnej formie

login : ….
hasło: ….
dane karty kredytowej: ….

3. należy zapisać jedynie skrót (hash) danych, a nie całą oryginalną treść

login : ….
hasło: ….
dane karty kredytowej: ….