



Instrukcja dla ucznia

1. Sprawdź, czy arkusz konkursowy z treścią zadań zawiera 7 stron. Ewentualny brak stron lub inne usterki zgłoś Komisji.
2. Sprawdź, czy w folderze konkursowym znajdują się pliki:
Znaczk1_30.in, znaczk2_30.in, znaczk1_500.in,
znaczk2_500.in, wyniki_ABC_30.out i wyniki_DE_30.out,
wyniki_ABC_500.out i wyniki_DE_500.out
3. Czytaj uważnie treść zadań.
4. Rozwiązania zadań zapisuj regularnie w folderze konkursowym na pulpicie komputera.
5. Pamiętaj o poprawnym nazywaniu plików z rozwiązaniami zadań (zgodnie z instrukcją w treści zadań).
6. Rozwiązując zadania zadbaj o uniwersalność rozwiązań tak, aby działały prawidłowo **na innych zestawach danych wejściowych**.
7. Rozwiązując zadania, możesz tworzyć różne pliki traktując je, jako brudnopis. Zapisy w brudnopisach nie będą sprawdzane i oceniane.
8. Po zakończeniu pracy z zadaniami, zgłoś ten fakt Komisji poprzez podniesienie ręki.
9. W obecności przedstawiciela Komisji upewnij się że wgrałeś na platformę konkursową pliki z rozwiązaniami oraz plik *metryczka.xls*

Etap Wojewódzki

2024 r.

Czas pracy:

120 minut

**Liczba punktów do
uzyskania:**

60 punktów

P O W O D Z E N I A



Organizacja danych w plikach

Plik z przykładowymi danymi wejściowymi: **znaczk1_30.in** i **znaczk2_30.in**.

Plik z danymi wyjściowymi: **wyniki_ABC_30.out** i **wyniki_DE_30.out**.

*Dodatkowy plik do testowania rozwiązań z większą liczbą danych: **znaczk1_500.in** i **znaczk2_500.in** (odpowiednio plik z danymi wyjściowymi: **wyniki_ABC_500.out** i **wyniki_DE_500.out**).*

Do oceny należy oddać plik zawierający w nazwie *Kod Ucznia oraz symbol zadania* (np. **K01_ABC.cpp** i **K01_DE.cpp**).

Opis organizacji danych w pliku wejściowym

W pierwszym wierszu pliku **znaczk1_30.in** zapisana jest jedna liczba naturalna **N** oznaczająca ilość liczb znajdujących się w kolejnym wierszu pliku. Liczba **N** mieści się w przedziale obustronnie domkniętym od **1** do **100 000**. W kolejnym wierszu zapisanych jest **N** liczb z zakresu od **1** do **1000** oddzielonych spacją. Liczby te stanowią dane do rozwiązania poniższych zadań. Każda liczba oznacza nominał kolejnego znaczka w klaserze pierwszym wyrażony w groszach.

W pliku **znaczk2_30.in** zapisanych jest **N** liczb z zakresu od **1** do **1000** oddzielonych spacją. Liczby te (razem z danymi z pliku **znaczk1_30.in**) stanowią dane do rozwiązania zadań C, D i E. Każda liczba oznacza nominał kolejnego znaczka w klaserze drugim wyrażony w groszach.

Przykład organizacji danych w plikach **znaczk1_30.in**:

30

769 70 80 97 17 3 5 16 46 78 95 51 34 25 49 63 139 59 526 34 67 20 25 22 34 45 57 63 75 82

Przykład organizacji danych w plikach **znaczk2_30.in**:

225 12 9 25 34 9 12 21 56 63 74 23 14 13 30 26 27 22 27 530 160 28 33 49 52 67 72 88 21 20

Korzystając z powyższych informacji napisz programy, które wykonają czynności wymienione w punktach A, B, C, D i E.



„Znaczki pocztowe”

Filatelistyka to hobby dające wiele satysfakcji. Kolekcjonerzy poszukują znaczków rzadkich, unikatowych, związanych z ważnymi wydarzeniami historycznymi. Co ciekawe, cenione są znaczki z widocznymi błędami i omyłkowym wydrukiem. Każdy znaczek jest drukiem ścisłego zarachowania, który posiada widoczną wartość nominalną (np. 80 gr.).

Adaś i Bartek od lat kolekcjonują znaczki. Każdego dnia do swoich kolekcji dokładają dokładnie jeden znaczek i co jakiś czas spotykają się, żeby wspólnie przejrzeć swoje kolekcje. Znaczki w klaserach chłopców ułożone są tematycznie, dlatego wartości nominalne kolejnych znaczków stanowią ciąg liczb nieuporządkowanych.

A. (7 punktów)

Pewnego dnia Adaś postanowił przyjrzeć się nominałom znaczków i pogrupować je w klaserze w ciekawy sposób. Na początek postanowił wybrać te, których cyfra dziesiątek ich wartości jest podzielna przez 2 lub przez 3.

Wczytaj liczby z pliku wejściowego **znaczkil_30.in** do odpowiednich struktur danych oraz **umieść w kodzie źródłowym w postaci komentarzy** nazwy zmiennych wraz z typami danych jakie zostały użyte do zapamiętania powyższych informacji. Wyświetl na ekranie i zapisz do pliku wyjściowego liczbę oznaczającą: **Ilość liczb spośród wczytanych, których cyfra dziesiątek jest podzielna przez 2 lub przez 3** (Pisząc program pamiętaj, że liczba zero dzieli się przez każdą liczbę z wyjątkiem zera).

Przykład

Dane wejściowe z pliku znaczkil_30.in:

30

769 70 80 97 17 3 5 16 46 78 95 51 34 25 49 63 139 59 526 34 67 20 25 22 34 45 57 63 75 82

Dane wyjściowe:

A: 22

B. (4 punkty)

Chłopcy przeglądając forum przeznaczone dla filatelistów dowiedzieli się, że obecnie panuje trend w poszukiwaniu znaczków z nominałem, którego suma cyfr jest większa od 10.

Dla wczytanego zestawu danych policz, ile takich nominałów jest w kolekcji przedstawionej w pliku znaczkil_30.in.

Przykład

Dane wejściowe z pliku znaczkil_30.in:

30

769 70 80 97 17 3 5 16 46 78 95 51 34 25 49 63 139 59 526 34 67 20 25 22 34 45 57 63 75 82

Dane wyjściowe:

B: 11



SZKÓŁ PODSTAWOWYCH – ETAP WOJEWÓDZKI

C. (8 punktów)

W ostatnim tygodniu na matematyce Adaś i Bartek poznali NWD. Dowiedzieli się, że największym wspólnym dzielnikiem dwóch liczb nazywamy największą liczbę naturalną, która dzieli obie te liczby bez reszty i nauczyli się ją wyznaczać. Postanowili porządnie przećwiczyć tę umiejętność z wykorzystaniem swoich kolekcji. Popołudnie upłynęło im na wyznaczaniu NWD dla kolejnych par znaczków.

Twoje zadanie będzie nieco trudniejsze, niż chłopców i będziesz potrzebować do tego dodatkowych informacji, dlatego wczytaj teraz liczby z pliku wejściowego **znaczk2_30.in** do odpowiednich struktur danych oraz umieść w kodzie źródłowym w postaci komentarzy nazwy zmiennych wraz z typami danych jakie zostały użyte do zapamiętania powyższych informacji. Dane z pliku **znaczk1_30.in** wciąż będą potrzebne.

Wyświetl na ekranie i zapisz do pliku wyjściowego liczbę oznaczającą: ile jest par kolejnych liczb z plików **znaczk1_30.in** oraz **znaczk2_30.in**, dla których największy wspólny dzielnik jest liczbą nieparzystą.

(UWAGA: Para liczb to np. pierwsza liczba z pliku **znaczk1_30.in** i pierwsza liczba z pliku **znaczk2_30.in** itd.)

Przykład

Dane wejściowe z pliku **znaczk1_30.in**:

30

769 70 80 97 17 3 5 16 46 78 95 51 34 25 49 63 139 59 526 34 67 20 25 22 34 45 57 63 75 82

Dane wejściowe z pliku **znaczk2_30.in**:

225 12 9 25 34 9 12 21 56 63 74 23 14 13 30 26 27 22 27 530 160 28 33 49 52 67 72 88 21 20

Dane wyjściowe:

C: 23

UWAGA

Punktacja konkursu uwzględnia złożoność obliczeniową programu.

Za poprawne wykonanie zadań D i E w złożoności obliczeniowej liniowej lub logarytmicznej otrzymujesz dodatkowe 13 punktów. Za poprawne wykonanie zadań z wykorzystaniem dodatkowych iteracji otrzymujesz dodatkowe 9 punktów.

Za zadanie D i E możesz otrzymać maksymalnie 41 punktów.

D. (18 punktów)

Powtarzając materiał do sprawdzianu z matematyki chłopcy zatrzymali się przy ciągach liczbowych, ponieważ od razu temat skojarzył im się ze znaczkami ułożonymi kolejno w klaserze. Stanowią one zdecydowanie ciąg liczb losowych, nie określony żadną regułą, ani funkcją, jednak można zauważyć fragmenty, w których kolejne nominały stale narastają, bądź przeciwnie – maleją. Ich zbiory doskonale nadają się do poszukiwania podciągów spójnych malejących i rosnących. Przeczytaj krótką informację, która pomoże Ci zrozumieć, czym są podciągi spójne i wykonać zadanie konkursowe:



SZKÓŁ PODSTAWOWYCH – ETAP WOJEWÓDZKI

Dowolny fragment ciągu liczb zachowujący kolejność występowania liczb nazywamy spójnym, jeśli jego elementy w ciągu wyjściowym występują obok siebie. Na przykład, ciąg liczb: **3,6,2** jest podciągiem spójnym ciągu wyjściowego 5,**3,6,2**,8,5

Zauważ, że jeśli kolejne liczby podciągu spójnego spełnią warunek, że każdy następny element jest większy bądź równy poprzedniemu, to otrzymamy **podciąg spójny niemalejący**.

Przykład:

Ciąg liczb: **2,5,8** jest podciągiem spójnym niemalejącym ciągu wyjściowego 6,**2,5,8**,1,7

Bartek i Adaś postanowili, że co jakiś czas w umówionym dniu rozpoczną układanie znaczków w klaserach tak, aby następny był z nominałem nie mniejszym od poprzedniego. Nie ustalili jednak, jak długie mają być ich ciągi niemalejących nominałów. Czy udało im się ułożyć ciągi o tej samej długości? Może nawet nie jeden raz, ciekawe jaki był najdłuższy ciąg niemalejący występujący w tym samym miejscu w obu klaserach?

Dla wczytanego zestawu danych **wyznacz maksymalną długość spójnych podciągów niemalejących, występujących w tym samym miejscu ciągów liczb w obu plikach** (wyznaczona maksymalna długość ma być wspólna dla obu ciągów, ciągi nie muszą składać się z takich samych elementów). Wyświetl na ekranie i zapisz do pliku wyjściowego:

- liczbę oznaczającą długość najdłuższego spójnego podciągu niemalejącego występującego jednocześnie w obu plikach,
- kolumnę zawierającą liczby stanowiące kolejne elementy poszukiwanego podciągu z pliku znaczk1_30.in,
- kolumnę zawierającą liczby stanowiące kolejne elementy poszukiwanego podciągu z pliku znaczk2_30.in.

Szukane podciągi na pewno są równej długości i w obu plikach występują dokładnie w tym samym miejscu ciągu (np. pomiędzy piątym, a dziesiątym elementem ciągu).

Przyjmij, że w badanych danych można wyznaczyć tylko jedną wartość maksymalną dla sytuacji, w której w tym samym miejscu w obu ciągach są spójne podciągi niemalejące o tej samej długości. Punktacja zadania uwzględnia złożoność obliczeniową programu.

Przykład:

Dane wejściowe z pliku znaczk1_30.in:

30

769 70 80 97 17 **3 5 16 46 78 95** 51 34 25 49 63 139 59 526 34 67 20 25 22 34 45 57 63 75 82

Dane wejściowe z pliku znaczk2_30.in:

225 12 9 25 34 **9 12 21 56 63 74** 23 14 13 30 26 27 22 27 530 160 28 33 49 52 67 72 88 21 20



Dane wyjściowe:

D:

6

3 9

5 12

16 21

46 56

78 63

95 74

E. (10 punktów)

Adaś przeglądał oba klasery zawierające znaczki ułożone zgodnie ze wcześniejszą umową. Zastanawiał się, jaki najdłuższy ciąg udało im się obu ułożyć w klaserach bez zwracania uwagi na to, czy występują w tych samych miejscach.

Dla wczytanego zestawu danych wyznacz **maksymalną długość spójnych podciągów niemających spełnioną jednocześnie dla danych w obu plikach** (wyznaczona maksymalna długość ma być wspólna dla obu ciągów liczb, wyznaczone podciągi nie muszą składać się z takich samych elementów, nie muszą występować w tych samych miejscach ciągów liczb). Wyświetl na ekranie i zapisz do pliku wyjściowego:

- liczbę oznaczającą długość najdłuższego spójnego podciągu niemającego,
- kolumnę zawierającą liczby stanowiące kolejne elementy poszukiwanego podciągu z pliku znaczki1_30.in,
- kolumnę zawierającą liczby stanowiące kolejne elementy poszukiwanego podciągu z pliku znaczki2_30.in,

Przyjmij, że w badanych danych można wyznaczyć najwyżej tylko jedną wartość maksymalną dla badanej sytuacji.

Punktacja zadania uwzględnia złożoność obliczeniową programu.

Przykład:

Dane wejściowe z pliku znaczki1_30.in:

30

769 70 80 97 17 3 5 16 46 78 95 51 34 25 49 63 139 59 526 34 67 20 25 **22 34 45 57 63 75 82**

Dane wejściowe z pliku znaczki2_30.in:

225 12 9 25 34 9 12 21 56 63 74 23 14 13 30 26 27 22 27 530 160 **28 33 49 52 67 72 88** 21 20

Dane wyjściowe:

E:

7

22 28

34 33

45 49

57 52

63 67

75 72

82 88

SZKÓŁ PODSTAWOWYCH – ETAP WOJEWÓDZKI

UWAGI DO ROZWIĄZAŃ

- Dane zawarte w przykładowych plikach wejściowych *znaczk1_30.in*, *znaczk2_30.in*, *znaczk1_500.in*, *znaczk2_500.in*, są poprawne (nie wymagają sprawdzania poprawności).
- W rozwiązaniach należy zadbać o odpowiednie dobranie struktur i typów danych do przechowywanych zawartości, poprawną inicjalizację zmiennych, czytelność i przejrzystość kodu i zastosowanie algorytmu o jak najmniejszej złożoności czasowej.
- Rozwiązania będą testowane i oceniane **na innych danych niż załączone do zadania** (zgodnych ze specyfikacją zadania).
- Do oceny należy przesłać jeden plik źródłowy o nazwie K01_ABC.cpp (z rozszerzeniem **cpp** lub **c**), realizujący rozwiązanie zadań z punktów od A do C, oraz drugi o nazwie K01_DE.cpp realizujący rozwiązanie zadań z punktów D i E. Nie należy załączać pliku wykonywalnego **exe**.
- Każdy program powinien składać się tylko z jednego pliku i korzystać jedynie ze standardowych bibliotek.
- Dane do programu powinny być pobierane z plików wejściowych. Można to wykonać poprzez **wczytanie danych z pliku wejściowego** lub poprzez **przekierowanie standardowego wejścia**.
- Rozwiązania nie mogą tworzyć plików tymczasowych, procesów czy wątków, korzystać z funkcji sieciowych, oczekiwać na interakcję użytkownika, uruchamiać innych programów, korzystać z zewnętrznych bibliotek (oprócz wymienionych powyżej).
- Przed rozwiązaniami punktów od A do E należy wyświetlić na ekranie odpowiednią literę określającą poszczególne części zadania.

Przykład informacji, jakie powinny pojawić się na ekranie po uruchomieniu programu:

Dla plików wejściowego *znaczk1_30.in*, *znaczk2_30.in*, dołączonych w folderze ucznia, programy powinny wyświetlić na ekranie następujące informacje:

- Odpowiedzi do zadań A, B, C w programie K01_ABC.cpp:

A: 20

B: 11

C: 23

- Odpowiedzi do zadań D, E w programie K01_DE.cpp:

D:

6

3 9

5 12

16 21

46 56

78 63

95 74

E:

7

22 28

34 33

45 49

57 52

63 67

75 72

82 88