

Ćwiczenia w optymalizacji rozwiązań algorytmicznych pod względem złożoności czasowej. Należy:

- 1) Zaimplementować 2 różne rozwiązania dla jednego z problemów, jedno z nich powinno mieć możliwie najniższy koszt czasowy
- 2) Wstawić liczniki operacji elementarnych w kodzie obu rozwiązań
- 3) Określić pesymistyczną złożoność czasową każdego rozwiązania
- 4) Ustalić, które z rozwiązań jest efektywniejsze
- 5) W rozwiązaniach tego zestawu nie można korzystać z gotowych kontenerów i algorytmów

Problem 1 – Samochody

Pewien uczciwy handlarz z dalekiego kraju przyjechał do Polski i zajmuje się sprzedażą samochodów. Posiada komis, z którego w ostatnim czasie zaczęły zniknąć auta. Handlarz pochodzi z kraju, gdzie zjawisko kradzieży jest niespotykane i stąd jego wielki smutek. Postanowił ogrodzić on teren komis. Z racji ostatnich strat finansowych jest on w stanie ogrodzić tylko pewien prostokątny fragment komis, ale za to taki, w którym stoją samochody o największej wartości. Jak go szybko wyznaczyć? Handlarz zwraca się do Ciebie z pomocą w tej sprawie.

Wejście:

W pierwszej linii wejścia podany jest rozmiar n na m ($1 \leq n, m \leq 1000$) prostokątnego placu. Na placu znajduje się m rzędów (liczba wierszy), a w każdym z nich po n samochodów (liczba kolumn). W kolejnych m liniach wejścia znajdują się liczby naturalne (po n w każdej linii) oznaczające ceny samochodów (liczby te nie przekraczają 1000000) na kolejnych miejscach. W ostatniej linii znajdują się liczby całkowite a i b ($1 \leq a \leq n, 1 \leq b \leq m$) oznaczające rozmiary prostokąta, który chce ogrodzić handlarz (b jest liczbą wierszy, a liczbą kolumn).

Wyjście:

W pierwszej linii wyjścia ma pojawić się liczba całkowita oznaczająca całkowitą wartość chronionych samochodów przy optymalnym ich ogrodzeniu, a w drugiej linii współrzędne lewego górnego rogu prostokąta (numer wiersza i kolumny, numeracja od jednego). Jeśli jest kilka takich prostokątów, to należy podać współrzędne tego o najmniejszym numerze wiersza, a w dalszej kolejności o najmniejszym numerze kolumny.

Przykład:

Wejście (in.txt):

```
5 4          //rozmiary placu
7 5 1 1 9    //ceny samochodów w pierwszym rzędzie placu
1 9 2 8 1    //w drugim itd.
5 9 9 2 6
3 8 3 4 8
3 2          //handlarz może ogrodzić prostokąt o rozmiarze 3 na 2
```

Wyjście (out.txt):

```
39          //maksymalna łączna wartość ogrodzonych samochodów
2 2         //ogrodzony prostokąt      9 2 8
                                           9 9 2
```

Problem 2 - Kody

Pan Gniewomir jest specjalistą od kodowania, ale ma awersję do komputerów, co jest zapewne spowodowane jego bardzo zaawansowanym wiekiem. Jego papierowa baza danych zawierająca różne kody rozszerza się jednak coraz bardziej i przestaje on nad tym panować. Nie starcza mu czasu, by wszystko ogarnąć, a i on chce czasem odpocząć i się rozerwać w klubie dla seniorów. Stąd potrzebna mu pomoc informatyka. Zaimplementuj algorytm, który będzie obliczał częstość występowania kodów binarnych. Długość kodu może wynosić maksymalnie 25 bitów, a każdy kod zaczyna się jedyneką.

Wejście:

W pierwszym wierszu wejścia podana jest liczba n ($1 \leq n \leq 10^6$) oznaczająca licznosc zbioru kodów. W kolejnych n wierszach wejścia podanych jest n kodów.

Wyjście:

W jedynej linii wyjścia ma się pojawić liczba wystąpień najczęściej występującego kodu w zbiorze.

Przykład:

Wejście (in.txt):

```
7          //7 kodów w zbiorze
10010      //kody ze zbioru
101
11100
1001
10010
101
10010
```

Wyjście (out.txt):

```
3          //jeden z kodów (10010) powtarza się aż 3 razy
```

Problem 3 – Misja

Ziemia weszła w konflikt z obcą cywilizacją, która posiada wyjątkowo niebezpieczną broń mogącą zniszczyć naszą planetę. Kapitan kosmicznego statku „Enterprise” ma za zadanie unieszkodliwić wroga. Posiada on rozmieszczenie baz nieprzyjaciela i planuje im przeszkodzić. W tym celu zostanie użyta nowatorska broń nadprzestrzenna, która niszczy wszystkie obiekty położone w zasięgu jej wiązki (jej szerokość kątowna to 90 stopni). Mając do dyspozycji współrzędne kątowe baz nieprzyjaciela należy wyznaczyć optymalny punkt ulokowania broni, by unieszkodliwić jak najwięcej baz za jednym wystrzałem. Dzięki swoim nadzwyczajnym umiejętnościom algorytmicznym możesz uratować ziemię!

Wejście:

W jedynej linii wejścia podana jest liczba n baz nieprzyjaciela ($1 \leq n \leq 10^6$). W kolejnych n liniach podane są po dwie liczby całkowite oznaczające współrzędne kątowe kolejnych baz (pierwsza liczba to stopnie od 0 do 359, a druga to minuty od 0 do 59) w stosunku do miejsca przetrzymywania broni (mogą się one powtarzać). Możemy założyć, że kąt 0 stopni to północ, 45 stopni to północny-wschód, 90 stopni to wschód itd. Jeden stopień składa się z 60 minut.

Wyjście:

W pierwszym wierszu wyjścia podana jest maksymalna możliwa liczba zniszczonych baz, jeśli broń będzie ustawiona pod odpowiednim kątem

Przykład:

Wejście (in.txt):

```
7          //7 baz
170 0      //pierwsza baza ma wsp. kątową 170 stopni i 0 minut
95 15      //itd.
0 5
260 0
70 23
190 0
330 38
```

Wyjście (out.txt):

```
3          //przy odpowiednim ustawieniu broni można zniszczyć bazy o wsp. 170, 190 i 260 st
```