

Zadanie 1.

W pewnym zamorskim kraju postanowiono ujednoczyć sposób ustalania cen na towary i usługi (podobna ma to uprościć podatki i zmniejszyć liczbę urzędników, ale ludzie mówią, że chodzi o to, że stosowny minister jest matematykiem z zacięciem informatycznym).

Przyjęto następujący sposób: ceny na towary będą liczbami "super pierwszymi", czyli takimi liczbami naturalnymi, które:

- są liczbami pierwszymi;
- suma cyfr tej ceny jest również liczbą pierwszą.

Cena na usługi musi spełniać trzeci warunek:

- oprócz wymienionych dwóch warunków, spełnia warunek trzeci: suma cyfr w jej zapisie binarnym jest także liczbą pierwszą.

Problem:

Napisz program, który dla danego przedziału cenowego $\langle a, b \rangle$, gdzie $a \geq 1$, $b \leq 10000000$, wyznaczy wszystkie możliwe ceny na usługi.

Wejście:

Na standardowe wejście należy podać dwie liczby całkowite znajdujące się w oddzielnych wierszach, pierwsza oznacza kraniec a przedziału poszukiwania cen na usługi, druga oznacza kraniec b przedziału poszukiwania cen na usługi.

Wyjście:

Na standardowe wyjście podawane są liczby całkowite będące cenami usług wyznaczonymi dla danego przedziału $\langle a, b \rangle$. Każda liczba powinna się znajdować w oddzielnym wierszu.

Przykład:

Zadanie 2.

Prezydent pewnego zamorskiego kraju postanowił rozwinąć dużą sieć komputerową serwerów oferujących różne usługi.

Sieć jest zbudowana z n serwerów połączonych dwukierunkowymi przewodami. Dwa serwery mogą być połączone bezpośrednio przez co najwyżej jeden przewód. Każdy serwer może być podłączony bezpośrednio do co najwyżej 10 innych serwerów i każde dwa serwery są połączone ze sobą pewną ścieżką w sieci. Każdy przewód ma stałą, wyrażoną wartością dodatnią, czas transmisji danych mierzony w milisekundach. Odległość (w milisekundach) $d(V, W)$ pomiędzy dwoma serwerami V i W , jest określona jako długość najkrótszej ścieżki (najkrótszy czas transmisji) łączącej V i W w sieci. Dla wygody niech $D(V, V) = 0$ dla wszystkich V .

Niektóre serwery oferują więcej usług niż inne. Dlatego każdy serwer V jest oznaczony liczbą naturalną $r(V)$, nazywaną rangą. Im wyższa ranga serwera, tym jest on wydajniejszy.

Na każdym serwerze mogą być przechowywane dane o pobliskich serwerach, ale nie o wszystkich. Wprowadzono zasadę, że nie są przechowywane dane o odległych serwerach o niskich rangach. Dokładniej, serwer W jest interesujący dla serwera V , gdy dla każdego serwera U takiego, że $D(V, U) \leq D(V, W)$, zachodzi $r(U) \leq r(W)$.

Na przykład, wszystkie serwery o maksymalnej randze są interesujące dla wszystkich serwerów. Jeśli serwer V ma maksymalną rangę, wówczas są dla niego interesujące serwery o maksymalnej randze. Niech $B(V)$ oznacza zbiór serwerów interesujących dla serwera V .

Chcemy obliczyć całkowitą ilość danych o serwerach, które muszą być przechowywane w sieci, będące sumą rozmiarów wszystkich zbiorów $B(V)$. Zamorski kraj oczekuje, że ilość danych będzie dość mała, tak, by zbudować sieć w taki sposób, aby suma ta nie przekraczała $30 \cdot n$.

Problem:

Napisz program, który:

- czyta opis sieci serwerów ze standardowego wejścia,
- wylicza całkowitą ilość danych o serwerów, która musi być przechowywana w sieci,
- zapisuje wynik na standardowe wyjście.

Wejście:

Wejście zaczyna się od liczby całkowitej z , która określa liczbę przypadków testowych. W kolejnych wierszach określone są poszczególne przypadki testowe.

Dla każdego testu w pierwszym wierszu podane są dwie liczby naturalne n , m , gdzie n oznacza liczbę serwerów w sieci ($1 \leq n \leq 30000$), a m oznacza liczbę kabli ($1 \leq m \leq 5n$). Liczby są oddzielone pojedynczym odstępem.

W kolejnych n wierszach podane są rangi serwerów. Linia i zawiera jedną liczbę r_i ($1 \leq r_i \leq 10$) - rangę i -tego serwera.

W kolejnych m wierszach opisane są połączenia. Każdy przewód jest opisany przez trzy liczby, a, b, t ($1 \leq t \leq 1000, 1 \leq a, b \leq n, a \neq b$), gdzie a i b oznaczają numery serwerów połączonych przewodem t i t jest czasem transmisji poprzez przewód w milisekundach.

Wyjście:

Na wyjście składa się liczba oznaczająca dla każdego przypadku testowego całkowitą ilość danych, które muszą być przechowywane w sieci.

Przykład:

Wejście:

```
1
4 3
2
3
1
1
1 4 30
2 3 20
3 4 20
```

Wyjście

```
9
```

Ponieważ $B(1) = \{1, 2\}$, $B(2) = \{2\}$, $B(3) = \{2, 3\}$, $B(4) = \{1, 2, 3, 4\}$.

Zadanie 3.

W pewnym zamorskim kraju prezydent zlecił swojemu ministrowi do spraw propagandy zadanie, polegające na opracowaniu szablonu, służącego do umieszczania na ścianach, plakatach i transparentach napisów opisujących piękno kraju w taki sposób, aby zajmowały one dużą powierzchnię. W celu zminimalizowania kosztów operacji (banany i inne owoce w tym roku nie obrodziły) minister musi opracować szablon.

Malując po szablonie, malujemy po nim całym, bez pomijania liter i ciągów liter w szablonie. Dopuszcza się możliwość, że któraś litera lub ciąg liter zostanie namalowana wielokrotnie (szablon ponownie przyłożony będzie nachodził na już napisany napis.). Litery na szablonie umieszczone są w jednym ciągu, bez spacji i innych znaków, nie będących literami. Możliwe jest wykonanie szablonu, który zawiera cały napis, ale prezydent chce zminimalizować koszty operacji i szablon powinien być jak najkrótszy.

Problem:

Napisz program, który:

- wczyta ze standardowego wejścia napis, który jest hasłem określonym przez prezydenta,
- obliczy długość minimalnego szablonu potrzebnego do wypisania hasła,
- wypisze wynik na standardowe wyjście.

Wejście:

W pierwszym i jedynym wierszu standardowego wejścia znajduje się jedno słowo, zawierające nie więcej niż 5000 liter. Minimalna długość słowa to jeden znak.

Wyjście:

W pierwszym i jedynym wierszu standardowego wyjścia należy zapisać liczbę całkowitą - długość szablonu.

Przykład:

Wejście:

ababbababbabababbabababbababbaba

Wyjście:

8

Ponieważ:

```

                                     a b a b b a b a
                                a b a b b a b a
                           a b a b b a b a
                      a b a b b a b a
                 a b a b b a b a
            a b a b b a b a
a b a b b a b a b b a b a b b a b a b b a b a b b a b a
```