Niech $G = (X, Y, E)$ będzie nieskierowanym grafem dwudzielnym (to znaczy $X \cap Y = \varnothing$ oraz dla każdego $\{x, y\} \in E, \{x, y\} \cap X \neq \varnothing$ i $\{x, y\} \cap Y \neq \varnothing$). Zbiór krawędzi $S \subseteq E$ nazywamy *skojarzeniem* w $G$ wtw, gdy żadne dwie krawędzie w $S$ nie są incydentne, tzn. dla każdych dwóch różnych krawędzi $e_1 \in S$ i $e_2 \in S$, $e_1 \cap e_2 = \varnothing$. Wierzchołek $x \in X \cup Y$ nazywamy *skojarzonym* względem $S$, gdy istnieje $y \in X \cup Y$, takie że $\{x, y\} \in S$. Skojarzenie $S$ nazywamy *doskonałym* wtw, gdy wszystkie wierzchołki z $X \cup Y$ są skojarzone względem $S$.

Załóżmy, że krawędzie $G$ są ważone wagami dodatnimi, tzn. zdefiniowana jest funkcja $w : E \to \mathbb{N} \setminus \{0\}$. Zadanie polega na napisaniu programu szukającego doskonałego skojarzenia w $G$ o maksymalnej wadze. Waga skojarzenia $S$ jest zdefiniowana następująco:

$$w(S) = \min\{w(e) : e \in S\},$$

czyli wagą skojarzenia jest waga najlżejszej krawędzi w nim zawartej. Uwaga! Nie jest to definicja standardowo przyjmowana.

Należy napisać program, który

- wczyta ze standardowego wejścia dwudzielny graf $G$,
- sprawdzi, czy w $G$ istnieje skojarzenie doskonałe i jeżeli tak, to zwróci wagę maksymalnego skojarzenia doskonałego w $G$,
- wypisze na standardowym wyjściu opis maksymalnego skojarzenia doskonałego w $G$.

## Wejście

1. Program czyta dane ze standardowego wejścia.
2. W pierwszym wierszu znajdują się dwie liczby całkowite $N$ i $M$ oddzielone spacjami, $1 \leqslant N \leqslant 500000$, $1 \leqslant M \leqslant 100000$. $M$ oznacza liczbę wierzchołków grafu $G$, $N$ oznacza liczbę krawędzi. Wierzchołki są numerowane od 1 do $M$.
3. Każdy z następnych $N$ wierszy zawiera po trzy liczby całkowite $x$, $y$, $w$ oddzielone spacjami ($1 \leqslant w \leqslant 1000000000$), oznaczające krawędź $\{x, y\}$ o wadze $w$, taką że $x \in X$ oraz $y \in Y$.

4. Jeśli dane wejściowe nie spełniają powyższej specyfikacji, program powinien wypisać na standardowy strumień błędów komunikat o błędzie i zakończyć działanie (np. w formacie ERR msg, gdzie msg to opis błędu). Można przyjąć, że na wejściu jest zawsze graf dwudzielny (tzn. pierwszy element każdej trójki opisującej krawędź należy do $X$, zaś drugi do $Y$).

## Wyjście

1. Program wypisuje wyniki na standardowe wyjście.
2. W pierwszym wierszu wyjścia powinna zostać wypisana jedna liczba całkowita – waga maksymalnego skojarzenia doskonałego w $G$ lub $-1$, jeżeli w $G$ nie ma skojarzenia doskonałego.
3. Jeżeli w $G$ istnieje skojarzenie doskonałe, to w kolejnych wierszach należy wypisać krawędzie wchodzące w skład skojarzenia doskonałego o maksymalnej wadze. Każdy z tych wierszy powinien zawierać trzy liczby całkowite $x$, $y$ i $w$ oddzielone spacjami, opisujące krawędź $\{x, y\}$ o wadze $w$ w taki sposób, że $x \in X$ oraz $y \in Y$.
4. Program może wypisać dowolne skojarzenie doskonałe z $G$, które ma maksymalną wagę.

## Ustalenia techniczne

Jako rozwiązanie należy dostarczyć plik `skojarzenie.cc`. Rozwiązanie zadania należy zdeponować w repozytorium w katalogu `$USERID/skojarzenie/`, gdzie `$USERID` to identyfikator konta studenckiego (inicjały plus numer indeksu). W pliku `$USERID/skojarzenie/zespol` należy zapisać identyfikatory osób tworzących parę (lub trójkę), która opracowała rozwiązanie. W rozwiązaniu zadania zalecane jest jak najszersze zastosowanie struktur danych i algorytmów oferowanych przez STL.

## Przykład

Dla danych wejściowych:

```
7 6
1 2 7
3 2 6
3 4 5
1 4 6
1 6 5
3 6 1
5 6 7
```

Twój program powinien wypisać:

```
6
1 4 6
3 2 6
5 6 7
```

## Bonus

Graf podany na wejściu nie musi być dwudzielny. Rozpoznając poprawność danych wejściowych należy sprawdzać czy graf jest dwudzielny. Jeżeli nie jest, należy wypisać odpowiedni komunikat i zakończyć działanie programu.

## Algorytm

Problem należy rozwiązać w oparciu o Algorytm 1 opisany poniżej. W ramach tego algorytmu wyszukiwane są skojarzenia doskonałe w podgrafach grafu wejściowego. Wyznaczania skojarzenia doskonałego w grafie należy zrealizować w oparciu o opisany poniżej Algorytmu 2. W ramach tego algorytmu wyznaczane są ścieżki rozszerzające w grafie wejściowym względem skojarzenia w tym grafie. *Ścieżka rozszerzająca* względem skojarzenia $S$ to ciąg wierzchołków $(v_0, \ldots, v_{2n-1})$, taki że $v_0$ jest nieskojarzony w $S$ oraz $v_{2n-1}$ jest nieskojarzony w $S$, dla każdego $0 \leqslant i < n$ $\{v_{2i}, v_{2i+1}\} \notin S$ i dla każdego $1 \leqslant i < n$ $\{v_{2i-1}, v_{2i}\} \in S$, oraz każdy wierzchołek grafu występuje na ścieżce co najwyżej raz. Innymi słowy, ścieżka rozszerzająca zaczyna się i kończy na nieskojarzonych wierzchołkach, składa się z krawędzi, które na przemian nie są i są w $S$ oraz nie zawiera cykli (w szczególności ścieżka złożona z jednej krawędzi łączącej dwa nieskojarzone wierzchołki jest rozszerzająca).

**Algorithm 1**: Szukanie skojarzenia doskonałego o maksymalnej wadze

```
Input: graf dwudzielny G = (X, Y, E), w_1, ..., w_m – wagi krawędzi
       G(E) uporządkowane rosnąco
if #X = #Y then
    /* #Z oznacza liczność zbioru Z                                  */
    l := 1;
    r := m + 1;
    while l < r do
        j := (l + r)/2;
        w := w_j;
        niech G_w podgraf G zawierający tylko krawędzie o wadze nie
        mniejszej niż w i wszystkie wierzchołki G;
        if w G_w istnieje skojarzenie doskonałe then
            l := j + 1;
        else
            r := j;
    /* jeżeli w G jest skojarzenie doskonałe, to w_{l-1} jest
       wagą skojarzenia doskonałego o maksymalnej wadze
       (wystarczy zwrócić dowolne skojarzenie doskonałe w
       G_{w_{l-1}})                                                   */
else
    G nie zawiera skojarzenia doskonałego;
```

**Algorithm 2**: Szukanie skojarzenia doskonałego

```
Input: graf dwudzielny G = (X, Y, E)
if G zawiera wierzchołki nienależące do żadnej krawędzi then
    /* w G nie ma skojarzenia doskonałego */
else
    /* jest szansa na skojarzenie doskonałe w G,
       konstruujemy początkowe skojarzenie */
    S := ∅;
    foreach x ∈ X do
        if istnieje nieskojarzone y, takie że {x, y} ∈ E then
            weź dowolne y jak wyżej;
            S := S ∪ {x, y};
    /* rozszerzamy S dopóki się da, szukając ścieżek
       rozszerzających */
    while istnieje wierzchołek nieskojarzony względem S i jest szansa
    na skojarzenie doskonałe w G do
        niech v_0 będzie dowolnym wierzchołkiem nieskojarzonym
        względem S;
        startując z v_0, przeszukujemy G w głąb, przechodząc na
        przemian krawędziami nie należącymi i należącymi do S w
        poszukiwaniu wierzchołka różnego od v_0 i nieskojarzonego
        względem S;
        if wierzchołek nieskojarzony nie został znaleziony then
            /* w G nie ma skojarzenia doskonałego */
        else
            niech v_{2n-1} będzie znalezionym wierzchołkiem, zaś
            (v_0, ..., v_{2n-1}) znalezioną ścieżką, która do niego prowadzi;
            /* (v_0, ..., v_{2n-1}) jest ścieżką rozszerzającą,
               rozszerzamy S usuwając z niego te krawędzie,
               które należą do ścieżki, a następnie dodając
               do niego te krawędzie ścieżki, które nie
               należą do skojarzenia */
            S := (S \ {{v_1, v_2}, ..., {v_{2n-3}, v_{2n-2}}}) ∪
            {{v_0, v_1}, ..., {v_{2n-2}, v_{2n-1}}};
/* Jeżeli w G jest skojarzenie doskonałe, to S jest takim
   skojarzeniem */
```