1. Bieg (25 punktów)

Zadanie

W krainie Galów, w której znajduje się V wiosek, odbywa się coroczny bieg pomiędzy dwiema oddalonymi wioskami. Po drodze w pewnych wioskach dostępny jest magiczny napój, którego wypicie skraca 10-krotnie czas potrzebny na przebycie pozostałej drogi do mety. Wypicie kolejnego napoju już nic nie zmienia. Asterix startuje w takich zawodach i chce zaplanować najszybsze pokonanie trasy.

Sieć wiosek reprezentowana jest jako nieskierowany graf ważony G, w którym wierzchołki to wioski, a krawędzie to drogi, którymi można się poruszać między wioskami. Graf G ma wierzchołki o numerach od 0 do $N-1$ i jest reprezentowany przez listę krawędzi. Każda krawędź to trójka w postaci u, v, w, gdzie $0 \leq u < v < V$ to numery wierzchołków, które łączą krawędź, a w to liczba minut, w jakich można przebyć drogę bez wspomagania się magicznym napojem. Czas jest wyrażony liczbą naturalną. Wierzchołki, których numer jest liczbą podzielną przez siedem, oznaczają wioski w których dostępny jest magiczny napój. Proszę napisać program, który zwraca najkrótszy czas, w jakim można przebyć z wioski o numerze s do wioski o numerze t.

Wejście

Pierwszy wiersz zawiera cztery liczby: $5 \leq V \leq 500$ będącą liczbą wierzchołków grafu, liczbę $5 \leq E \leq 5000$ będącą liczbą krawędzi w grafie, liczby $0 < s < t < V$ będące numerami wiosek, gdzie znajdują się start i meta biegu. Kolejne E wierszy zawiera trójki w postaci u, v, w, gdzie u i v to numery wierzchołków ($u < v$), które łączą, a w to liczba minut, w jakich można przebyć drogę z u do v bez użycia napoju.

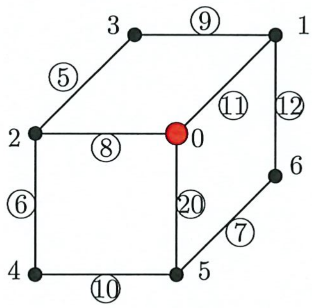
Wyjście

W jedynym wierszu standardowego wyjścia należy umieścić liczbę będącą minimalnym czasem przebycia drogi wyrażoną w minutach z dokładnością do jednego miejsca po kropce, np. 30.0, 29.6.

Przykład

Dla danych wejściowych:

```
7 9 4 6
0 1 11
0 2 8
0 5 20
1 3 9
1 6 12
2 3 5
2 4 6
4 5 10
5 6 7
```



Poprawną odpowiedzią jest: 16.3

Najszybsza droga wiedzie przez wioski: $4, 2, 0, 1, 6$, czas wynosi $6 + 8 + 11/10 + 12/10 = 16.3$

2. Ciąg (25 punktów)

Zadanie

Dany jest ciąg liczb pierwszych $C_1, C_2, C_3, ..., C_N$. Proszę napisać program, który sprawdza, czy jest możliwe pocięcie ciągu na kawałki w taki sposób, aby sumy elementów kolejnych kawałków stanowiły rosnący ciąg geometryczny o długości co najmniej 3 elementów. W przypadku istnienia więcej niż jednego podziału spełniającego warunki zadania, należy wybrać ten, który tworzy ciąg geometryczny o mniejszym ilorazie. Program powinien wypisać pierwsze dwa wyrazy znalezionego ciągu. W przypadku, gdy taki podział nie jest możliwy, program powinien wypisać słowo BRAK.

Wejście

Pierwszy wiersz zawiera liczbę $3 \leqslant N \leqslant 100$ wyrazów ciągu. Kolejne N wierszy zawiera ciąg N liczb pierwszych z przedziału $[2 - 10^{18}]$ będących kolejnymi wyrazami ciągu.

Wyjście

W jedynym wierszu standardowego wyjścia należy umieścić dwie liczby rozdzielone spacją, będące pierwszym i drugim wyrazem utworzonego ciągu geometrycznego. W przypadku, gdy podział nie jest możliwy, program powinien wypisać słowo BRAK.

Przykład

Dla danych wejściowych:

```
10
3
5
2
3
7
13
5
2
2
23
```

Poprawną odpowiedzią jest:

```
8 12
```

Powstały ciąg geometryczny ma postać:

```
8 12 18 27
```

3. Skok (25 punktów)

Zadanie

Trener skoku wzwyż chce się przekonać jaką maksymalną wysokość jest w stanie przeskoczyć zawodnik. Zakładamy, że poprzeczka może być ustawiona maksymalnie na H cm, a skoczek ma do dyspozycji maksymalnie N nieudanych prób (zrzutek). Napisz program, który obliczy minimalną liczbę skoków, jakie w **najgorszym** przypadku musi oddać zawodnik by wyznaczyć maksymalną wysokość, jaką potrafi pokonać.

Założenia:

1. Poprzeczkę można ustawić na dowolną całkowitą wysokość od 1 do H.
2. Jeżeli zawodnik nie pokona wysokości to traci próbę, jeżeli przeskoczy to może skakać dalej
3. Jeśli skoczek zrzuci poprzeczkę na wysokości h, to zrzuci ją również na każdej wysokości większej niż h.
4. Jeżeli skoczek przeskoczy poprzeczkę na wysokości h to pokona także każdą wysokość mniejszą niż h.

Wejście

Pierwszy i jedyny wiersz standardowego wejścia zawiera dwie liczby całkowite, $150 \leqslant H \leqslant 250$ (maksymalna wysokość poprzeczki) i $2 \leqslant N < 20$ (maksymalna liczba zrzutek).

Wyjście

Pierwszy i jedyny wiersz standardowego wyjścia zawiera dokładnie jedną liczbę całkowitą – minimalną liczbę skoków, jakie w **najgorszym** przypadku musi oddać zawodnik by wyznaczyć maksymalną wysokość, jaką potrafi pokonać.

Przykład

Dla danych wejściowych:

```
200 2
```

Poprawną odpowiedzią jest:

```
20
```

4. Woda (25 punktów)

Zadanie

Dane jest N pojemników na wodę o pojemnościach wyrażonych w litrach i będących liczbami naturalnymi $C_1 > C_2 > C_3 >, ..., > C_N$. Pierwszy pojemnik jest napełniony wodą do pełna, a pozostałe pojemniki są puste. Wodę możemy przelewać pomiędzy pojemnikami na dwa sposoby: albo napełniamy docelowy pojemnik do pełna, albo całkowicie opróżniamy pojemnik. Naszym celem jest odmierzenie jednego litra wody. Cała woda musi pozostać w pojemnikach. Proszę napisać program, który wylicza, ile minimalnie wody należy łącznie przelać, aby odmierzyć jeden litr wody. Do dyspozycji mamy maksymalnie 8 przelewań. Jeżeli odmierzenie jednego litra nie jest możliwe, program powinien wypisać słowo BRAK.

Wejście

Pierwszy wiersz zawiera liczbę $3 \leqslant N \leqslant 6$ pojemników. Drugi wiersz zawiera N liczb naturalnych $1 <= C_i <= 100$ będących pojemnościami kolejnych pojemników.

Wyjście

W jedynym wierszu standardowego wyjścia należy umieścić liczbę będącą minimalną, łączną ilością wody, jaką należało przelać, aby w jednym z pojemników znalazł się jeden litr wody. W przypadku, gdy takie odmierzenie nie jest możliwe, program powinien wypisać słowo BRAK.

Przykład

Dla danych wejściowych:

```
4
20 12 7 2
```

Poprawną odpowiedzią jest:

```
15
```

Stan pojemników po kolejnych przelaniach miał postać:

```
(18, 0, 0, 2) przelano 2 litry,
(18, 0, 2, 0) przelano 2 litry,
(16, 0, 2, 2) przelano 2 litry,
(16, 0, 4, 0) przelano 2 litry,
(14, 0, 4, 2) przelano 2 litry,
(14, 0, 6, 0) przelano 2 litry,
(12, 0, 6, 2) przelano 2 litry,
(12, 0, 7, 1) przelano 1 litr,
```

W ostatnim pojemniku pozostał 1 litr wody. Razem przelano 15 litrów. Rozwiązanie można uzyskać także napełniając do pełna pojemniki o pojemnościach 12 i 7 litrów, ale wymaga to przelania łącznie 19 litrów wody.