

注意事項

1. 本次競賽共有 6 題，每題滿分皆為 100 分，共 600 分。本次競賽的總得分為各題的得分總和。
計算排名時，以總得分由高至低排序為主，若有同分的狀況，則以最後一次分數變動的時間點為基準，較早者名次較高。
2. 本次競賽使用 CMS 系統。CMS 系統說明網址：<https://reurl.cc/EZ7qxa>。
3. 本次競賽每題皆由數個子任務（子題）構成，每個子任務中有數筆測試資料，需完全答對某一子任務中的所有測試資料才能獲得該子任務的分數。單次上傳得分為所有子任務得分的總和，該題得分為每個子任務得到的最大得分的總和。
例如，假設某一題有兩個子任務。某人傳了兩份程式，第一份程式只成功解決第一個子任務，第二份程式只成功解決第二個子任務，則這個人得到的分數為兩個子任務的分數和。
4. 本次競賽不得參考任何線上資料、隨身碟等電子式的資料，系統附的「STL 標準樣板函數庫」不在此限。可以參考紙本資料。
在比賽過程中禁止瀏覽本系統、「STL 標準樣板函數庫」以外之網頁。請勿嘗試以任何方法破壞本系統。
5. 若需要裝水、上廁所等，請通知監考人員。
6. 本次競賽不提供即時計分板。
7. 作答的小技巧
 - 題目要看清楚，重要的條件不要漏看。
 - 題目有很多很簡單子任務，當想不到整體題目怎麼解決時，可以先想子任務拿到部分分數。
 - 做好時間分配，適當的把整場比賽的時間分配到每一題上面。
 - 對於使用 C++ 輸入輸出（cin、cout）的參賽者，由於題目輸入較多，為改善輸入輸出之效能，請在 `int main()` 的第一行插入以下程式碼，並且請勿同時使用 C 輸入輸出（scanf、printf、puts）。
`ios_base::sync_with_stdio(0); cin.tie(0);`

祝比賽順利！

第一題：分糖果 (candy)

問題敘述

今天是延平幼兒園的零嘴日，準備的是大家最愛的糖果！

延平幼兒園總共有 n 位小朋友，他們的座號分別是 $1, 2, \dots, n$ 。每位小朋友手上都有不同數量的糖果，其中座號為 i 的小朋友手上有 a_i 顆糖果。分完糖果後，老師發現每位小朋友手上的糖果數量不同，這可能會引發嫉妒和紛爭。為了解決這個問題，老師提出了一種方法，盡量平均分配每個人的糖果。方法如下：

1. 首先，老師將小朋友們按照座號順時針排成一圈，如下圖所示。
2. 然後，老師會指向座號為 1 的小朋友，嘗試平均分配他和他旁邊（順時針方向）的小朋友手上的糖果。如果這兩人手上的糖果總數是偶數，則他們各自拿一半的糖果；如果是奇數，則下一位小朋友會多拿一顆糖果。
3. 接著，老師指向順時針下一位小朋友，並重複執行步驟 2。這個過程會重複 k 次。

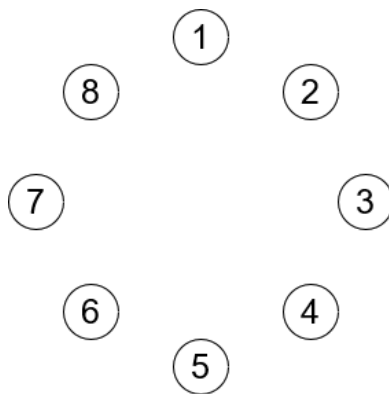


Figure 1: $n = 8$ 時的示意圖。

經過以上步驟，整個平分的流程就完成了。現在，給定小朋友的數量 n ，每位小朋友一開始手上的糖果數量 a_1, a_2, \dots, a_n ，以及重複執行的次數 k ，你能夠寫一個程式來計算經過平分流程後，每位小朋友手上有多少顆糖果嗎？

輸入格式

單筆輸入。輸入第一行含有兩個整數 n 和 k 。

輸入第二行含有 n 個整數 a_1, a_2, \dots, a_n ，數字之間以一個空白隔開。

輸出格式

輸出 n 個整數於一行，第 i 個整數代表座號為 i 的小朋友在平分流程後會有幾顆糖果。

測資限制

- $2 \leq n \leq 100$ 。
- $0 \leq k \leq 100$ 。
- $0 \leq a_i \leq 100$ ($1 \leq i \leq n$)。

輸入範例 1

```
6 0
1 2 3 4 5 6
```

輸出範例 1

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

輸入範例 2

```
5 3
12 14 10 9 12
```

輸出範例 2

```
13 11 10 11 12
```

輸入範例 3

```
5 100
84 37 19 9 35
```

輸出範例 3

```
37 36 37 37 37
```

備註

在輸入範例 1 中，因為 $k = 0$ ，所以每個小朋友手上的糖果數量都不會變動。

在輸入範例 2 中，平分的流程如下：

- 首先座號為 1 和 2 的小朋友會平分兩人手上的糖果，平分後兩人各拿到 13 顆，每個小朋友的糖果數量分別是 [13, 13, 10, 9, 12]。
- 接著座號為 2 和 3 的小朋友會平分兩人手上的糖果。因為兩人總共有 23 顆糖果，不能均分，根據規則編號為 3 的小朋友會多拿一顆。這時每個小朋友的糖果數量分別是 [13, 11, 12, 9, 12]。
- 最後座號為 3 和 4 的小朋友會平分兩人手上的糖果。兩人總共有 21 顆糖果，同樣不能均分，根據規則編號為 4 的小朋友會多拿一顆。這時每個小朋友的糖果數量分別是 [13, 11, 10, 11, 12]。至此就完成所有流程。

評分說明

本題共有 6 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	0	範例測試資料。
2	15	$k = 0$ 。
3	20	$n = 2$ ，且在 k 次平分中，兩人手上糖果數量和都是偶數。
4	30	$n = 2$ 。
5	20	在 k 次平分中，兩人手上糖果數量和都是偶數。
6	15	無額外限制。

第二題：食物組合 (combination)

問題敘述

延平中學的校慶園遊會是每位學生一年中最期待的活動之一，因為這個活動提供了各種美食品嘗的機會！然而，由於每個人的胃容量有限，不可能品嚐所有攤位上的美食，因此需要謹慎選擇要品嚐的食物。

今年，延平中學共有 n 個攤位，攤位編號分別為 $1, 2, \dots, n$ ，每個攤位都售賣一種食物。經過仔細調查後，你發現攤位編號為 i 的攤位所提供的食物好吃程度為 a_i （由於一些攤位可能提供一些特殊或不太美味的食物，這個值可能是負數）。

由於個人胃口有限，你打算品嚐恰好 k 個攤位的食物。品嚐完後，你的滿足程度會是所選攤位的食物好吃程度的總和。基於你在數學課上學到的知識，你知道選擇 k 個攤位共有 C_k^n 種不同的選擇組合。你是否能計算出這 C_k^n 種組合中每一種的滿足程度？

輸入格式

單筆輸入。輸入第一行含有兩個整數 n, k ，代表攤位的數量及你想要造訪的攤位數量。

輸入第二行含有 n 個整數 a_1, a_2, \dots, a_n ，第 i 個數字 a_i 代表攤位編號為 i 的食物好吃程度。數字之間以一個空白隔開。

輸出格式

輸出 C_k^n 個整數，代表從 n 個攤位中選擇 k 個攤位的食物來吃，所有可能得到的滿足程度。將所有滿足程度由小到大輸出。

測資限制

- $1 \leq k \leq n \leq 50$ 。
- $1 \leq C_k^n \leq 10^5$ 。
- $-10^7 \leq a_i \leq 10^7$ ($1 \leq i \leq n$)。

輸入範例 1

```
5 3
1 2 3 4 5
```

輸出範例 1

6 7 8 8 9 9 10 10 11 12

輸入範例 2

6 2
-3 5 7 -2 -10 8

輸出範例 2

-13 -12 -5 -5 -3 -2 2 3 4 5 5 6 12 13 15

輸入範例 3

50 50
0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

輸出範例 3

0

備註

$C_k^n = \frac{n!}{k!(n-k)!}$ ，其中 $a! = a \times (a-1) \times \cdots \times 2 \times 1$ 代表 1 到 a 的所有正整數的乘積。特別地，我們定義 $0! = 1$ 。

在輸入範例 1 中，所有可能的組合如下：

- {1, 2, 3}，總和是 6。
- {1, 2, 4}，總和是 7。
- {1, 2, 5}，總和是 8。
- {1, 3, 4}，總和是 8。
- {1, 3, 5}，總和是 9。
- {2, 3, 4}，總和是 9。
- {1, 4, 5}，總和是 10。
- {2, 3, 5}，總和是 10。
- {2, 4, 5}，總和是 11。
- {3, 4, 5}，總和是 12。

評分說明

本題共有 5 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	0	範例測試資料。
2	17	$k = 1$ 。
3	32	$k \leq 4$ 。
4	35	$n \leq 20$ 。
5	16	無額外限制。

第三題：搭公車 (bus)

問題敘述

在延平中學，經常可以遇到來自新北、基隆、桃園等外縣市通勤的學生，其中使用公車或客運作為通勤方式相當普遍。

David 也是這樣的學生之一。他需要搭乘 n 趟公車才能從家到達延平中學，經過 $n + 1$ 個公車站。為了方便描述，我們將這些公車站編號為 $1, 2, \dots, n + 1$ 。David 的家位於公車站 1，而延平中學則位於公車站 $n + 1$ 。在這個問題中，我們假設從 David 家到公車站 1 以及從公車站 $n + 1$ 到延平中學的時間可以忽略不計。

David 的通勤需要轉乘 n 次不同的公車路線。我們將這 n 條路線編號為 $1, 2, \dots, n$ ，其中第 i 條路線將從公車站 i 出發，行駛至公車站 $i + 1$ ，需要花費 t_i 單位時間。每條路線都有 m_i 個發車時間，分別在時間點 $s_{i,1}, s_{i,2}, \dots, s_{i,m_i}$ 。抵達公車站 $i + 1$ 的時間則分別是時間點 $s_{i,1} + t_i, s_{i,2} + t_i, \dots, s_{i,m_i} + t_i$ 。轉乘所需的時間可以忽略不計；換句話說，當 David 到達第 i 個公車站時，只要第 i 條路線的發車時間不小於前一條路線的抵達時間，他就可以順利轉乘。

由於這漫長的通勤過程，David 對於怎麼樣才不遲到有點困擾。他設想了 q 種情況。在第 i 個情況中，David 設想如果 he 需要在時間點 d_i 或之前抵達學校，那他最晚需要在什麼時間點從家裡出發才不會遲到？David 已經給你 n 條公車路線的資訊和他設想的 q 種狀況，請你寫出一個有效率的程式算出每一種情況的答案，你能幫忙他嗎？

輸入格式

單筆輸入。輸入第一行含有兩個整數 n, q ，代表公車路線的數量和 David 設想的情況數量。

接下來 $2n$ 行會依序給出 n 條公車路線的資訊，每兩行代表一條公車路線。每條公車路線第一行有兩個整數 m_i, t_i ，代表有幾種抵達時間，以及這條路線的搭乘時間。第二行含有 m_i 個整數 $s_{i,1}, s_{i,2}, \dots, s_{i,m_i}$ 代表這 m_i 種到達第 i 個公車站的時間點，數字之間一個空白隔開。

最後一行含有 q 個整數 d_1, d_2, \dots, d_q 代表 David 設想的 q 種情況。

輸出格式

輸出 q 個整數於一行，第 i 個整數代表 David 設想的第 i 種情況的答案。如果不論怎麼搭乘都會遲到，輸出 -1 。

測資限制

- $1 \leq n, q \leq 3 \times 10^5$ 。

- $1 \leq m_i \leq 3 \times 10^5$ ($1 \leq i \leq n$)。
- $\sum m_i \leq 3 \times 10^5$ ，其中 $\sum m_i$ 代表所有 m_i 的總和。
- $1 \leq t_i \leq 10^9$ ($1 \leq i \leq n$)。
- $1 \leq s_{i,j} \leq 10^9$ ($1 \leq i \leq n$ 、 $1 \leq j \leq m_i$)。
- 同一條公車路線給出的 $s_{i,j}$ 會嚴格遞增，也就是 $s_{i,1} < s_{i,2} < \dots < s_{i,m_i}$ 。
- $1 \leq d_i \leq 10^9$ ($1 \leq i \leq q$)。

輸入範例 1

```
2 6
4 2
4 6 10 13
4 3
7 8 17 19
8 9 12 10 20 21
```

輸出範例 1

```
-1 -1 6 4 13 13
```

輸入範例 2

```
3 5
1 1000
1
5 5
100 200 300 400 500
5 10
600 700 800 900 1000
1 10 100 1000 10000
```

輸出範例 2

```
-1 -1 -1 -1 -1
```

備註

在輸入範例 1 中，如果 d_i 是 8 或 9（第 1 和第 2 個情況），那 David 不論如何都會遲到。但如果 $d_i = 10$ （第 4 個情況），David 就可以在時間點 4 從公車站 1 出發，在時間點 6 抵達公車站 2，並且在公車站 2 於時間點 7 坐上第 2 條路線，於時間點 10 抵達學校。而當 $d_i = 12$

時（第 3 個情況），David 可以在時間點 6 從公車站 1 出發，在時間點 8 抵達公車站 2，並且在公車站 2 於時間點 8 坐上第 2 條路線（即使發車時間和前一條路線的抵達時間恰好相同，David 也是可以轉乘的到），於時間點 11 抵達學校。

在輸入範例 2 中，因為第 1 條路線的搭乘時間過久，David 不論如何都會遲到。

評分說明

本題共有 6 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	0	範例測試資料。
2	16	$n = 1, q \leq 1000, m_1 \leq 1000$ 。
3	19	$n \leq 1000, q \leq 1000, \sum m_i \leq 1000$ 。
4	24	$n = 1$ 。
5	18	$m_1 = m_2 = \dots = m_n$ 。
6	23	無額外限制。

第四題：電力供給 (power-supply)

問題敘述

有一天，延平中學突然發生了停電事件！為了不中斷教學活動，學校必須立即啟用備用電源和預先準備好的電路設備。

學校共有 n 個房間，包括教室和辦公處，都需要恢復供電。這些房間編號為 $1, 2, \dots, n$ 。備用電源位於房間 1，這個房間不會斷電，並能夠提供其他房間所需的電力。

學校事先設置了一些電路以便應對不時之需。已知總共有 m 條電路，編號分別為 $1, 2, \dots, m$ 。每條電路連接兩個不同的房間，其功能在於當一個房間有電力時，可以供應另一個房間的電力需求。在 m 條電路中，第 i 條電路連接了房間 u_i 和 v_i 。這 m 條電路的佈局滿足以下條件：只要所有電路都沒有損壞，那麼位於房間 1 的備用電源可以直接或間接（通過一個或多個電路）供應所有房間的電力需求。如果將這個情況視為一個圖論問題，其中每個房間都是一個節點，每條電路都是無向邊，那麼這個圖形是一個連通圖。例如下圖就滿足這個條件，因為所有房間都可以從房間 1 透過一或多條電線抵達。例如，房間 2, 3, 4, 5 分別可以由路線 $1 \rightarrow 2$ 、 $1 \rightarrow 3$ 、 $1 \rightarrow 2 \rightarrow 4$ 和 $1 \rightarrow 2 \rightarrow 4 \rightarrow 5$ 抵達。

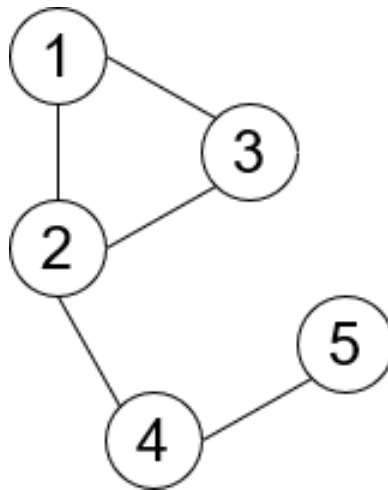


Figure 2: $n = 5$ 間房間時，一種電線可能的設置示意圖。每個圓圈代表一個房間，而圓圈之間的連線代表一條電線。

不幸的是，在停電事件中，這 m 條電路全部損壞了，需要電力公司的維修。這些電路分別歸屬於 k 家不同的電力公司之一，編號從 1 到 k 。每家電力公司的維修服務需要支付一個固定的基本費用 p_i 元，然後可以維修任意多條由該公司負責的電路。但如果要修復第 i 條電路，則需要支付額外的維修費用 w_i 元。

你的任務是以最低的成本，確保所有房間都能夠通過位於房間 1 的備用電源，直接或間接地獲得所需的電力供應。你是否能設計一個程式，計算出達到這個目標所需的最小費用？

輸入格式

單筆輸入。輸入第一行含有三個整數 n, m, k ，代表房間的數量、電路的數量和公司的數量。

第二行含有 k 個整數 p_1, p_2, \dots, p_k ，第 i 個整數 p_i 代表第 i 家公司的基本維修費用。

接下來 m 行，第 i 行含有四個整數 u_i, v_i, w_i, c_i ，分別代表第 i 條電路連接哪兩間房間、該條線路的額外維修費用及歸屬於哪一家公司負責。

輸出格式

輸出一個整數，代表如果要讓所有房間都能透過位於房間 1 的備用電源獲得電力供應，至少需要多少費用。

測資限制

- $2 \leq n \leq 1000$ 。
- $n - 1 \leq m \leq 5 \times 10^5$ 。
- $1 \leq k \leq 16$ 。
- $1 \leq p_i \leq 10^9$ ($1 \leq i \leq k$)。
- $1 \leq u_i, v_i \leq n$ ， $u_i \neq v_i$ ($1 \leq i \leq m$)。
- $1 \leq w_i \leq 10^9$ ($1 \leq i \leq m$)。
- $1 \leq c_i \leq k$ ($1 \leq i \leq m$)。
- 保證房間 1 可以藉由一或多條電路抵達其他所有房間。

輸入範例 1

```
4 7 3
7 3 5
1 3 1 3
2 4 1 2
2 4 1 3
3 4 5 1
1 2 4 1
1 4 10 3
3 1 2 2
```

輸出範例 1

```
17
```

輸入範例 2

```
6 5 6
987654321 988776655 938198371 988736281 938193712 992817394
1 4 100 1
4 2 200 2
3 4 300 3
1 6 400 4
4 5 500 6
```

輸出範例 2

```
4896184522
```

輸入範例 3

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

輸出範例 3

```
24
```

輸入範例 4

```
5 7 1
123
1 2 4 1
1 3 3 1
1 4 6 1
1 5 5 1
5 3 2 1
2 4 5 1
3 4 1 1
```

輸出範例 4

133

備註

輸入範例 1 中，學校可以選擇修復第 2、5 和第 7 條電線，這樣需要支付基本費用 $7 + 3 = 10$ 元，額外的費用 $1 + 4 + 2 = 7$ 元，總共是 17 元。如此一來，其他房間都能透過一或多條電線與房間 1 連通。

輸入範例 2 中，需要選擇所有電線，而且需要支付公司 1, 2, 3, 4, 6 的基本費用。

評分說明

本題共有 6 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	0	範例測試資料。
2	18	$m = n - 1$ 。
3	13	$k = 1$ 。
4	13	$k \leq 8$ 。
5	30	$k \leq 12$ 。
6	26	無額外限制。

第五題：區間中位數 (median)

問題敘述

現在有 n 個正整數 a_1, a_2, \dots, a_n 。我們一共有 $\frac{n(n+1)}{2}$ 張卡片，每張卡片上都寫下了一對正整數 l, r ，滿足 $1 \leq l \leq r \leq n$ ，每張卡片上面寫的正整數對都不相同。接下來，我們需要對所有卡片進行排序，排序的依據如下：

1. 首先，我們按照卡片上區間的中位數，從小到大進行排序。具體來說，如果一張卡片上的兩個數字是 l_i 和 r_i ，我們會從數列 a 中選擇第 l_i 到第 r_i 個數，也就是 $a_{l_i}, a_{l_i+1}, \dots, a_{r_i}$ 這些數字中的中位數。
2. 如果區間中位數相同，則按照卡片上的第一個數字，從小到大排序。
3. 如果以上兩個條件都相同，則按照卡片上的第二個數字，從小到大排序。

需要特別注意的是，在這個問題中，計算中位數的方法與一般的不同：具體來說，如果有 m 個數字，當 m 是奇數時，中位數是排序後的中間數，即第 $\frac{m+1}{2}$ 小的數字。如果 m 是偶數，中位數則是排序後的中間兩個數中較小的那個，即第 $\frac{m}{2}$ 小的數字。

這個問題的內容如下：給定一個介於 1 到 $\frac{n(n+1)}{2}$ 之間的整數 k ，請問你是否能夠找出排序後的第 k 張卡片是哪張，以及該卡片的區間中位數是多少？

輸入格式

單筆輸入。第一行有兩個整數 n 和 k 。

第二行含有 n 個整數 a_1, a_2, \dots, a_n ，數字之間以一個空白隔開。

輸出格式

輸出三個正整數於一行，分別代表第 k 張卡片的區間中位數及上面寫的兩個整數。

本題擁有特殊計分方式，請參考評分說明。

測資限制

- $1 \leq n \leq 2 \times 10^5$ 。
- $1 \leq k \leq \frac{n(n+1)}{2}$ 。
- $1 \leq a_i \leq n$ ($1 \leq i \leq n$)。

輸入範例 1

5 5
3 5 1 4 2

輸出範例 1

2 3 5

輸入範例 2

5 11
3 5 1 4 2

輸出範例 2

3 1 4

輸入範例 3

6 6
1 2 3 4 5 6

輸出範例 3

2 2 3

輸入範例 4

4 8
2 2 1 4

輸出範例 4

2 2 2

備註

輸入範例 1 和 2 中，我們可以算出所有區間的中位數，歸類如下：

- 區間中位數是 1：(2, 3)、(3, 3)、(3, 4)。
- 區間中位數是 2：(2, 5)、(3, 5)、(4, 5)、(5, 5)。
- 區間中位數是 3：(1, 1)、(1, 2)、(1, 3)、(1, 4)、(1, 5)。
- 區間中位數是 4：(2, 4)、(4, 4)。
- 區間中位數是 5：(2, 2)。

所有的卡片的排序就是 (2, 3)、(3, 3)、(3, 4)、(2, 5)、(3, 5)、(4, 5)、(5, 5)、(1, 1)、(1, 2)、(1, 3)、(1, 4)、(1, 5)、(2, 4)、(4, 4)、(2, 2)。

評分說明

本題共有 5 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	0	範例測試資料。
2	20	$n \leq 100$ 。
3	30	$n \leq 3000$ 。
4	10	$a_i = i$ ($1 \leq i \leq n$)。
5	40	無額外限制。

在這題當中，輸出的三個數字分別占 40%、30% 和 30% 該子任務的分數。舉例來說，如果你的程式能正確求出第一個值，但是無法求出後兩個值的話，依舊可以獲得 40% 該子任務的分數。**注意到，即使你只想要獲得每個子任務的部分分數，為了評分方便，你依舊需要輸出恰好三個整數**（比方說，你可以在後兩個數字都輸出 0）。

第六題：典型區間詢問問題 (range-query)

問題敘述

區間詢問類型的題目是在競程中一個很常見的題目類型。以下是 ChatGPT 對於這類型問題的介紹：

Range query problems are a common and fascinating category of problems in competitive programming. These problems often involve querying and manipulating data within specific intervals or ranges, requiring contestants to devise efficient algorithms and utilize appropriate data structures to handle a large number of queries or modifications.

Key characteristics of range query problems include:

- **Interval Operations:** In these problems, you are tasked with a series of operations or queries on elements within specific intervals. Operations may include interval modifications (e.g., increasing or decreasing values within a range) or interval queries (e.g., finding the maximum, minimum, or sum of values within a given range).
- **Variety of Data Structures:** To support interval operations, you must choose or implement suitable data structures. Common data structures used in these problems include segment trees, Fenwick trees (BIT), Fenwick trees on trees, and more. Each data structure has its advantages and limitations, and selecting the right one depends on the problem's characteristics.
- **Time Complexity:** Competitive programming often imposes strict time constraints. Hence, it's crucial to ensure that the time complexity of your interval operations is sufficiently low to handle large datasets efficiently. This may involve optimization techniques such as block decomposition, binary search, and more.
- **Offline vs. Online:** Some problems can be categorized as offline or online. In offline problems, you can preprocess interval operations before executing them in a certain order to obtain the final answers efficiently. In online problems, the operations occur in real-time, requiring you to respond promptly to queries.

Range query problems challenge contestants not only to understand data structures and algorithms but also to creatively tackle a variety of operations within different contexts. By studying and practicing these problems, you can improve your competitive programming skills and become better equipped to handle the challenges presented during competitions.

經過以上介紹後，在這題當中，我們同樣要解決一個典型的區間詢問類型的題目：現在有 n 個正整數 a_1, a_2, \dots, a_n 以及 q 次詢問，每次詢問會給出三個正整數 l_i, r_i, x_i ，請算出以下總和：

$$\sum_{j=l_i}^{r_i} (x_i \bmod a_j)$$

其中 $x_i \bmod a_j$ 代表 x_i 除以 a_j 的餘數。

輸入格式

單筆輸入。輸入第一行含有兩個整數 n 和 q ，代表序列的長度和詢問的數量。

輸入第二行含有 n 個正整數 a_1, a_2, \dots, a_n ，數字之間以一個空白隔開。

接下來 q 行，每行代表一筆詢問。每一行都有三個正整數 l_i, r_i, x_i 。

輸出格式

依序對於每筆詢問輸出一行，代表每筆詢問的答案。

測資限制

- $1 \leq n, q \leq 2 \times 10^5$ 。
- $1 \leq a_i \leq 2 \times 10^5$ ($1 \leq i \leq n$)。
- $1 \leq l_i \leq r_i \leq n$ ($1 \leq i \leq q$)。
- $1 \leq x_i \leq 2 \times 10^5$ ($1 \leq i \leq q$)。

輸入範例 1

```
5 5
3 17 6 9 10
2 4 100
1 3 6
2 5 23
4 4 87
3 4 31
```

輸出範例 1

```
20
6
19
6
5
```

輸入範例 2

```
4 5
3 1224 33 3290
1 4 32442
1 4 200000
1 4 132589
1 4 183028
1 4 99999
```

輸出範例 2

```
3453
3110
1415
2741
2163
```

輸入範例 3

```
6 4
33892 4000 4567 200000 28910 193029
3 6 133929
1 5 174920
1 6 193828
5 6 199382
```

輸出範例 3

```
287633
186134
243205
32275
```

備註

在輸入範例 1 中，每筆詢問的內容如下：

- $(100 \bmod 17) + (100 \bmod 6) + (100 \bmod 9) = 20$ 。
- $(6 \bmod 3) + (6 \bmod 17) + (6 \bmod 6) = 6$ 。
- $(23 \bmod 17) + (23 \bmod 6) + (23 \bmod 9) + (23 \bmod 10) = 19$ 。

- $(87 \bmod 9) = 6$ 。
- $(31 \bmod 6) + (31 \bmod 9) = 5$ 。

評分說明

本題共有 7 組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	0	範例測試資料。
2	8	$n, q \leq 3000$ 。
3	14	x_i ($1 \leq i \leq q$) 在 q 筆詢問中最多只有 50 種不同的值。
4	18	$l_i = 1, r_i = n$ ($1 \leq i \leq q$)。
5	11	$a_i \leq 50$ ($1 \leq i \leq n$)。
6	25	$a_i \geq 4000$ ($1 \leq i \leq n$)。
7	24	無額外限制。