

Énoncé

Les enfants Laura et Franck visitent entre amis une usine de confiserie spécialisée dans la guimauve — la vraie, extraite de la plante guimauve ! Ils découvrent comment sont fabriquées leur friandises préférées et goûtent plusieurs types de guimauves. À la fin de leur visite, ils décident de passer à la boutique de l'usine pour en ramener à leur maison.

Dans la boutique, il y a `nbGuimauves` guimauves différentes, et la $i^{\text{ème}}$ guimauve étant vendue au prix `prix[i]`. Laura et Franck décident d'en acheter au total `nbAchats`, mais comme ils aimeraient varier les plaisirs, ils ne vont pas prendre plusieurs fois la même guimauve.

Franck a trouvé un moyen de répartir le coût à la caisse avec son amie Laura :

- Si la guimauve coûte moins que `prixLimite`, alors c'est Laura qui la paye ;
- Sinon, Laura paye `prixLimite`, et Franck paye le reste, c'est-à-dire `prix[i] - prixLimite`.

Étant très sûr de sa logique mathématique, Franck refuse tout autre système de répartition. Notons `coutLaura` le montant total que va payer Laura et `coutFranck` le montant total que doit compléter Franck. Laura, étant très insatisfaite de cette répartition, et voulant se venger, souhaiterait minimiser la différence `diffCouts = coutLaura - coutFranck` **qui peut être négative**. Comme Franck est assez hésitant sur la quantité à ramener, Laura aimerait connaître la valeur minimale de `diffCouts` pour `nbRequetes` requêtes donnant `prixLimite` et `nbAchats`.

Donnez à Laura pour chacune de ses `nbRequetes` la plus petite valeur de `diffCouts` possible.

Entrée

- La première ligne contient deux entiers : `nbGuimauves` et `nbRequetes` ($1 \leq \text{nbGuimauves}, \text{nbRequetes} \leq 10^5$)
- La deuxième ligne contient `nbGuimauves` entiers : `prix[1] prix[2] ... prix[n]` ($1 \leq \text{prix}[i] \leq 10^9$).
- Suivent `nbRequetes` lignes, la y -ème d'entre elle contient deux entiers : `prixLimite[y]` et `nbAchats[y]` ($1 \leq \text{prixLimite}[y] \leq 10^9$; et $1 \leq \text{nbAchats}[y] \leq \text{nbGuimauves}$)

Attention, dans cet exercices les entiers utilisés peuvent être grand et **dépasser la capacité d'un entier 32 bits**. Il est recommandé d'utiliser des entiers 64 bits, avec le type `long long int`.

Sortie

Vous devez afficher `nbRequetes` lignes contenant un seul entier : `diffCout[z]` pour la $z^{\text{ème}}$ requête.

Score

Les tests de ce sujet sont répartis en 3 sous-tâches. Pour obtenir les points d'une sous-tâche, il vous faut valider tous ses tests.

- Sous-tâche 0 (0 pts) : Exemples
- Sous-tâche 1 (30 pts) : `nbGuimauves, nbRequetes` ≤ 1000 ; `prix[i], prixLimite[j]` $\leq 10^6$.

- Sous-tâche 2 (30 pts) : $\text{prixLimite}[1] = \dots = \text{prixLimite}[\text{nbRequetes}]$
- Sous-tâche 3 (40 pts) : Pas de contrainte supplémentaire.

Limites

Limite de temps : 1000 ms.

Limite de mémoire : 64000 kb.

Exemples

Exemple 1

```
5 2
1 9 22 10 19
18 4
5 2
```

La sortie attendue est :

```
34
-21
```

Dans sa première requête, Laura peut prendre les guimauves avec les prix 1, 9, 22 et 10. Laura paiera alors 38 et Franck 4. La réponse est donc $38 - 4 = 34$.

Dans la seconde requête, Laura prendra les guimauves avec les prix 22 et 19. Dans ce cas, elle paiera 10, alors que Franck devra payer 31. La réponse est $10 - 31 = -21$.

Exemple 2

```
7 4
1 5 4 3 7 11 9
5 4
5 7
7 3
4 5
```

La sortie attendue est :

```
4
16
7
1
```

Exemple 3

3 3
5 6 7
10 1
5 3
3 3

La sortie attendue est :

5
12
0