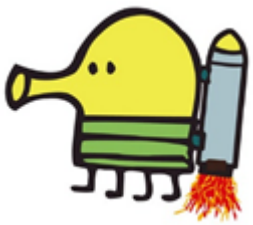
Lëtzebuerger Informatiksolympiad 2018

Finale

Tous les programmes doivent être réalisés sous forme d'applications console.
Les formats des données ainsi que des résultats sont à respecter absolument.
Sous le fichier d'entrée on entend soit l'entrée directe via le clavier
soit la redirection d'un fichier texte en mode console.

TÂCHE 1 DOODLE JUMP 50 points

Description



Dans le jeu Doodle Jump, un personnage nommé « The Doodler », initialement sur le sol, doit monter un escalier pour atteindre la ligne d'arrivée située à une altitude de **A** mètres au-dessus du sol. Chaque marche de l'escalier a une hauteur d'un mètre. Pour l'aider, **N** jet-packs sont répartis sur certaines des marches de l'escalier. Le **i**-ème jet-pack est placé à une hauteur de $\mathbf{h_i}$ mètres et contient $\mathbf{c_i}$ unités de carburant. Un jet-pack propulse le personnage sans effort $\mathbf{c_i}$ marches plus haut. Lorsque le carburant est épuisé, le personnage perd son jet-pack et doit continuer à monter l'escalier normalement. Le personnage ne peut jamais redescendre, seulement monter et il ne peut utiliser qu'un seul jet-pack à la fois. Il est garanti qu'un jet-pack ne dépasse jamais la ligne d'arrivée. Le personnage étant paresseux, il aimerait connaître le nombre minimal de marches qu'il est obligé de monter lui-même en optimisant le choix des jet-packs.

Tâche

Écrivez un programme qui calcule le nombre minimal de marches que le personnage doit monter sans l'aide de jet-packs.

Entrée et sortie du programme

Entrée

- La première ligne contient deux entiers **N** et **A**, le premier désignant le nombre de jet-packs et le second la hauteur en mètres de la ligne d'arrivée.
- Les **N** lignes suivantes contiennent chacune deux entiers $\mathbf{h_i}$ et $\mathbf{c_i}$, désignant respectivement la hauteur en mètres à laquelle le jet-pack se trouve et la quantité de carburant contenue dans cet **i**-ème jet-pack.

Sortie

- Le programme affiche un seul entier, le nombre minimal de marches que le personnage doit monter sans l'aide de jet-packs.

Restrictions

- $0 \leq \mathbf{N} \leq 10^5$
- $0 \leq \mathbf{A} \leq 10^9$
- Pour $1 \leq \mathbf{i} \leq \mathbf{N}$ on a : $0 \leq \mathbf{h_i}, \mathbf{c_i} \leq \mathbf{A}$ ainsi que $0 \leq \mathbf{h_i} + \mathbf{c_i} \leq \mathbf{A}$

Exemple d'exécution 1

Données (entrée)
3 6 2 2 1 2 2 3
Résultats (sortie)
3

Le personnage choisit de ne pas utiliser le premier jet-pack qu'il croise. Il prend le second qui l'emmène de la hauteur 2 à la hauteur 5 (3 mètres plus haut). Il monte la dernière marche en sautant. Il a donc monté 2 + 1 = 3 marches.

Exemple d'exécution 2

Données (entrée)
4 10 9 1 1 3 0 4 4 1
Résultats (sortie)
4

Le personnage utilise tous les jet-packs sauf celui à la hauteur 1.

Remettez le programme sous le nom DOODLEJUMP.xxx, avec xxx=PAS, C(PP), PY ou JAVA.

TÂCHE 2 INDEX 50 points

Tâche

On vient de trouver un manuscrit bizarre utilisant nos caractères usuels mais qui ne respecte pas notre ordre alphabétique. Heureusement, de ce document il subsiste l'index, qui est une liste alphabétiquement ordonnée. Votre travail consiste à trouver l'ordre alphabétique utilisé dans ce livre.

Entrée et sortie du programme

Entrée

Le fichier d'entrée contient une liste de mots composés de lettres 'A' à 'Z'. Un mot a une taille maximale de 40 lettres. La fin du fichier est indiquée par le caractère '#'. Le fichier ne contient pas d'incohérences !

Sortie

Le programme affiche une ligne contenant tous les caractères présents dans l'index et placés dans l'ordre alphabétique croissant utilisé dans ce livre mystérieux.

Exemple d'exécution

Données (entrée)
ION ANA ADONIA DOINA DOINN DDAN DDAO #
Résultats (sortie)
IANOD

Remettez le programme sous le nom INDEX.xxx, avec xxx=PAS, C(PP), PY ou JAVA.