



Take away

by Christian Kauth

Dear hc² contestants, let me welcome you to this third edition of a programming contest that gained considerable popularity and counts around 110 participants this year! Fantastic! My name is Heidi, the coding cow, and I am the mascot of the event. On this one special day a year, I use to travel... through Switzerland in 2010, around the world in 2011 and through the space-time continuum today! Generally I get stuck in some dicey situations, but I am confident you can help me solve the problems and get my adventurous journey continued.

To motivate you in doing so, we have some take-aways for you. Besides the refreshing drinks, tasty sandwiches, and stylish t-shirts, we have medals, balloons and prizes. Your share in the three last elements depends on your performance at the contest today. To maximize your profit, be clever, write proper code and act as an organized team. And even more importantly : Have fun!

Before the real contest starts, for which they have prepared 5 problems consisting of 3 subtasks each, the organizers would like to take advantage of your extraordinary coding skills to write the rewarding algorithm.

Easy Task - Medals

Upon solving a subtask correctly, your team receives 1 point and gets access to the next subtask, of advanced difficulty. For any incorrect submission (Compiler-Error, Timelimit, Run-Error, No-output, Wrong-answer, Presentation-Error), you pocket a penalty point. Also does your team possess a time accumulator. Each time you receive a point for a correctly solved subtask, this timer is immediately incremented by the actual contest time. The ranking is done first by points (more is better). Ties are broken via penalty points (less is better). Remaining ties are broken by the timer (less accumulated time is better). No further tie breakers need to be considered for this dry-run task.

You are to write a function `void find_medalists(int N, int p[], int n[], int t[])` that

- takes as arguments
 - N : the number of teams in the contest ($3 \leq N \leq 50$)
 - p : $p[i]$ holds the points of team i ($0 \leq p[i] \leq 15$)
 - n : $n[i]$ holds the penalty points of team i ($0 \leq n[i] \leq 100$)
 - t : $t[i]$ holds the accumulated time of team i in minutes ($0 \leq t[i] \leq 4050$)
- calls the three functions
 - `void gold_medalist(int yourGoldMedalist)`
 - `void silver_medalist(int yourSilverMedalist)`
 - `void bronze_medalist(int yourBronzeMedalist)`exactly once. `yourGoldMedalist` must be the index of the hc² 2012 champion, `yourSilverMedalist` the one of the vice-champion and `yourBronzeMedalist` the index of the team on rank 3.



À Emporter

par Christian Kauth

Chers participants au hc², permettez-moi de vous souhaiter la bienvenue à cette troisième édition d'un concours de programmation qui a gagné largement en visibilité et compte cette année près de 110 participants ! Super !

Mon nom est Heidi et je suis la mascotte de cet évènement. En ce jour très spécial de l'année, j'ai beaucoup voyagé ... à travers la Suisse en 2010, autour du monde en 2011 et aujourd'hui, grâce à votre aide, à travers le continuum espace-temps ! J'ai pour habitude de me retrouver, malgré moi, coincée dans des situations peu confortables, mais je suis certaine que vous pourrez m'aider à résoudre les problèmes et à reprendre mon voyage aventureux.

Pour vous motiver à le faire, nous avons quelques petites choses à emporter pour vous. Au-delà des boissons rafraîchissantes, des sandwiches gouteux et des super t-shirts, nous avons des médailles des ballons et des prix. Votre part des trois derniers éléments dépendra de votre performance au concours d'aujourd'hui. Pour maximiser vos profits, soyez malins, écrivez du code élégant et agissez en équipe. Plus important encore, *amusez-vous* !

Avant le début du vrai concours, pour lequel ils vous ont préparé 5 problèmes composés de 3 tâches chacun, les organisateurs voudraient traire bénéfice de vos incroyables talents de programmeurs afin d'écrire l'algorithme qui récompensera les meilleurs d'entre vous.

Tâche Facile - Médailles

En résolvant cette tâche correctement, votre équipe reçoit un point et obtient l'accès à la prochaine tâche, d'une plus grande difficulté. Pour toute soumission incorrecte (Compiler-Error, Timelimit, Run-Error, No-output, Wrong-answer, Presentation-Error), vous encaisserez un point de pénalité. De plus, votre équipe dispose d'un accumulateur de temps. Chaque fois que vous recevez un point pour une tâche correctement résolue, cet accumulateur est incrémenté du temps qui s'est écoulé depuis le début du concours. Le classement est fait en premier lieu avec les points (plus on en a, mieux on est classé). Les égalités sont départagées via les points de pénalité (moins on en a, mieux on est classé). Les égalités restantes sont départagées grâce à l'accumulateur de temps (moins de temps accumulé on a, mieux on est classé). Il n'y a pas besoin de considérer d'autres moyens de départager les égalités dans cette tâche.

Vous devez écrire une fonction `void find_medalists(int N, int p[], int n[], int t[])` qui

- prend comme arguments
 - N : le nombre d'équipes dans le concours ($3 \leq N \leq 50$)
 - p : $p[i]$ contient le nombre de points de l'équipe i ($0 \leq p[i] \leq 15$)
 - n : $n[i]$ contient le nombre de points de pénalité pour une équipe i ($0 \leq n[i] \leq 100$)
 - t : $t[i]$ contient le temps accumulé pour l'équipe i en minutes ($0 \leq t[i] \leq 4050$)
- appelle les trois fonctions
 - `void gold_medalist(int yourGoldMedalist)`
 - `void silver_medalist(int yourSilverMedalist)`
 - `void bronze_medalist(int yourBronzeMedalist)`exactement une fois. `yourGoldMedalist` doit être l'indice du champion hc² 2012, `yourSilverMedalist` celui de l'équipe médaillée d'argent et `yourBronzeMedalist` l'indice de l'équipe ayant gagné la médaille de bronze.



Zum Mitnehmen

von Christian Kauth

Liebe hc² TeilnehmerInnen, ich heisse euch herzlich Willkommen zur dritten Auflage eines Programmierwettbewerbs der stets an Popularität gewonnen hat und heute rund 110 Teilnehmer zählt! Das ist fantastisch! Mein Name ist Heidi, die programmierende Kuh, und ich bin das Maskottchen des Events. An diesem jährlichen Ausnahmetag, habe ich es mir zur Gewohnheit gemacht zu verreisen... durch die Schweiz in 2010, um die Welt in 2011 und durch das Raum-Zeit-Kontinuum heute! Dabei verstricke ich mich des öfteren in brenzlige Situationen. Aber ich zähle auf euch, mir zu helfen meine Probleme zu lösen, damit ich meine abenteuerliche Odyssee fortsetzen kann.

Zum Ansporn haben wir einige Mitnimmsel für euch bereit. Nebst den erfrischenden Getränken, den köstlichen Brötchen und der stylischen Tshirts, haben wir Medaillen, Ballons und Preise. Euer Anteil an den drei letzten Elementen hängt von eurer Leistung während des Wettbewerbs ab. Um euern Gewinn zu erhöhen, seid raffiniert, schreibt korrekten Code und agiert als Team. Und noch wichtiger : Habt Spass dabei!

Der Wettbewerb enthält 5 Aufgabenstellungen, bestehend aus je 3 Unteraufgaben. Bevor es aber los geht, möchten die Veranstalter noch von euern hervorragenden Programmierfähigkeiten Gebrauch machen um den Belohnungsalgorithmus zu schreiben.

Einfache Aufgabe - Medaillen

Für jede korrekt gelöste Unteraufgabe erhaltet ihr einen Punkt und Zugriff auf die nächste, schwierigere Unteraufgabe. Für eine fehlerhafte Lösung (Compiler-Error, Timelimit, Run-Error, No-output, Wrong-answer, Presentation-Error), kassiert ihr einen Strafpunkt. Auch besitzt euer Team einen Zeitsammler. Jedesmal wenn ihr mit einer korrekten Lösung einen Punkt sammelt, wird dieser Zeitsammler unverzüglich um die aktuelle Wettbewerbszeit erhöht. Das Klassement beruht zuerst auf den Punkten (mehr ist besser). Gleichstände werden dank der Strafpunkte gebrochen (weniger ist besser). Alle restlichen Gleichstände werden durch den Zeitsammler gebrochen (weniger ist besser). Für diese Aufwärmübung braucht ihr keine weiteren Kriterien zum Brechen der Gleichstände in Betracht zu ziehen.

Schreibt eine Funktion `void find_medalists(int N, int p[], int n[], int t[])` welche

- folgende Parameter nimmt
 - N : die Anzahl der Teams im Wettbewerb ($3 \leq N \leq 50$)
 - $p : p[i]$ enthält die Punkte von Team i ($0 \leq p[i] \leq 15$)
 - $n : n[i]$ enthält die Strafpunkte von Team i ($0 \leq n[i] \leq 100$)
 - $t : t[i]$ enthält die insgesamt gesammelte Zeit von Team i in Minuten ($0 \leq t[i] \leq 4050$)
- die folgenden drei Funktionen
 - `void gold_medalist(int yourGoldMedalist)`
 - `void silver_medalist(int yourSilverMedalist)`
 - `void bronze_medalist(int yourBronzeMedalist)`jeweils einmal aufruft. `yourGoldMedalist` muss der Index des hc² 2012 Champions sein, `yourSilverMedalist` der des Vize-Champion und `yourBronzeMedalist` der Index des Drittplatzierten.