

Spies & Marmots

by Christian Kauth

When the first warming sun rays of spring hit the icy landscapes, and snowmelt starts pulling plashy creeks through the Earth's dazzling snow coat, then time has come for Heidi to set out to the mountains to awake the marmots from their hibernation. A thrilling year of challenging projects of unthinkable space-, time- and cyber-travel awaits these diligent rodents. Needless to say that Heidi is the head of these endeavours and she kept thinking while her fellow employees rested. She realized that the marmots' productivity could be tremendously boosted by discarding spy marmots, which look like marmots but are counterproductive scapegraces.

So Heidi decides to keep a list of hard-working marmots. To do so, she scans the marmots at work and adds motivated workers to the list. Marmots which are not yet on the list, are considered harmful spies for the time being. An intricate communication channel allows Heidi to share list updates with the entire marmot community. Hence each time a marmot meets another rodent, it instantly knows whether its *via-à-vis* is an engaged buddy or a spying scapegrace. This is very important, because Heidi's motto to success is never to engage into a project with a spy! You shall help Heidi and the marmots to set up this antispyware.

Write Heidi's function `scan_marmot` that adds a marmot with a given identification number `id` to the list of commonly known hard workers. This function will be called N times and it may use the function `send_hint` at most M times in total to send each time *one byte* of information over the channel to the marmots. The marmots then receive this same byte of information via the function `receive_hint`, which you shall code for them. You are to encode the transmitted information on hardworking marmot IDs in a way that the marmots' function `is_spy`, may correctly answer queries on the nature of rodents, based on their ID. It must return whether the marmot is already on the list or not. $M = 2 \cdot N$ for all subtasks and only Heidi may obtain the value of N by calling `get_N`. Note that `scan_marmot` cannot directly call `receive_hint` nor share memory with this function, as Heidi and the marmots live in two different processes since she got lost in cyberspace a year ago. Any interaction must happen through calls to `send_hint`.

Easy Task

Information must be shared immediately, as calls to `is_spy` may interlace calls to `scan_marmot`. It is guaranteed that $0 \leq id < 2^{16}$ and $0 \leq N \leq 2^{16}$.

Sample Run

Function calls	Expected output	Explanation
<code>scan_marmot(15)</code>		Adds 15 to the list
<code>is_spy(15)</code>	<code>false</code>	15 is on the list, hence not a spy
<code>is_spy(3)</code>	<code>true</code>	3 is assumed to be a spy, as not on the list
<code>scan_marmot(3)</code>		Adds 3 to the list
<code>is_spy(2014)</code>	<code>true</code>	2014 is not on list, hence assumed a spy
<code>is_spy(3)</code>	<code>false</code>	we know now that 3 is not a spy

Espions & Marmottes

par Christian Kauth

Quand les premiers rayons de soleil chauffent le paysage enneigé, et la fonte des neiges crée des ruisseaux clapoteux au travers du manteau si blanc de la Terre, c'est l'heure pour Heidi de monter haut dans les montagnes pour réveiller les marmottes de leur hibernation. Une année palpitante de projets ambitieux de voyages dans l'espace, le temps et même le cyberspace, va commencer pour ces rongeurs diligents. Il va de soi que Heidi est à la tête de ces projets, qu'elle a élaborés pendant que ses employés se reposaient. Elle a réalisé que la productivité des marmottes pourrait être augmentée considérablement en écartant les espions, qui ressemblent à des marmottes, mais sont en fait des gouapes contreproductives.

C'est ainsi que Heidi décide d'établir une liste de marmottes appliquées. A cette fin elle scanne toutes les marmottes au travail et ajoute celles qui ont l'air motivé à la liste. Les marmottes qui actuellement ne figurent pas (encore) sur la liste sont considérées comme espions nocifs. Un canal de communication élaboré permet à Heidi de transmettre les mises-à-jour de la liste à l'entière communauté des marmottes. Ainsi, toutes les fois qu'une marmotte croise un autre rongeur, elle sait immédiatement si son vis-à-vis est un collègue engagé ou une gouape. Cette connaissance est primordiale dans le nouveau concept de Heidi : Ne jamais se lancer dans un projet avec un espion ! A vous d'aider Heidi et les marmottes à monter cette plateforme anti-espionnage.

Ecris pour Heidi une fonction `scan_marmot` qui ajoute une marmotte avec un certain numéro d'identification `id` à la liste des bosseurs. Cette fonction sera appelée N fois et elle peut utiliser la fonction `send_hint` au maximum M fois pour à chaque appel envoyer *un byte* d'information aux marmottes via le canal. Les marmottes reçoivent ce même byte d'information par la fonction `receive_hint`, que tu devras coder pour elles. Tu dois encoder l'information sur les identités (ID) transmises de sorte à ce qu'un appel de la fonction `is_spy` des marmottes, que tu dois programmer aussi, permette de déterminer si le rongeur en question figure déjà sur la liste ou non. $M = 2 \cdot N$ pour toutes les sous-tâches et seule Heidi a accès à la valeur N via la fonction `get_N`. Note que `scan_marmot` ne peut pas directement appeler `receive_hint` ou partager de la mémoire avec, puisque Heidi et les marmottes habitent dans deux processus différents depuis qu'elle s'est perdue dans le cyberspace il y a une année. Toute interaction se passe via `send_hint`.

Tâche Facile

Les mises à jour doivent être communiquées instantanément. Les appels à `is_spy` et `scan_marmot` peuvent être mélangés. Il est garanti que $0 \leq id < 2^{16}$ et $0 \leq N \leq 2^{16}$.

Exemple d'exécution

Appels de fonction	Sortie attendue	Explication
<code>scan_marmot(15)</code>		ajoute 15 à la liste
<code>is_spy(15)</code>	<code>false</code>	15 est sur la liste, donc pas un espion
<code>is_spy(3)</code>	<code>true</code>	3 est suspect, car pas sur la liste
<code>scan_marmot(3)</code>		ajoute 3 à la liste
<code>is_spy(2014)</code>	<code>true</code>	2014 est suspect, car pas sur la liste
<code>is_spy(3)</code>	<code>false</code>	en fait, 3 n'est pas un espion