

Maya extension

by Christian Kauth

Heidi feels very uncomfortable about the Maya Long Count calendar, chiseled in stone and ending on bak'tun 13, equivalent to our December 21st 2012. If the time and space came to an end by that day, not only would there never be an hc² 2013, but also could Heidi not risk to use her belting time machine to travel into a distant future... So she decided to pay a visit to the Mayas. State-of-the-art lithography will find enough space on the calendar rock to extend it to an unguessable future!

The calendar is hidden inside a vast stepped pyramid of which Heidi owns a map. The pyramid contains R ($1 \leq R \leq 1000$) cylindrical rooms and from each room, up to C ($1 \leq C \leq 10$) bidirectional corridors leave to other rooms, but never back to the same room again! Also are there no two corridors connecting the same pair of rooms. The entrances to these corridors are astonishingly regularly arranged along the rooms' circumference and upon entering a room, the mere view of this symmetry will not help Heidi to orient herself. Luckily, a map contains this information. For each room i , labeled from 1 to R , is given an list of up to C labels, indicating the destinations of the tunnels leaving the room, sorted by the counter-clockwise order of their entrance.

The list for room 3 in Figure 1 is (4, 5, 1, 2). Entering it from room 1 and picking the second counter-clockwise exit, would lead Heidi to room 4. Coming from room 2, she would have to take the first exit to get to room 4. The fourth exit brings Heidi back to her previous room in this case!

Let's name the outside of the pyramid room 0. There is exactly one entrance/exit to the pyramid, i.e. there is one bidirectional tunnel connecting room 0 to any other room. The calendar is hidden in treasure room R . Help Heidi, starting from outside the pyramid (room 0) to get into the treasure room (room R), where she upgrades the calendar, and again out of the pyramid.

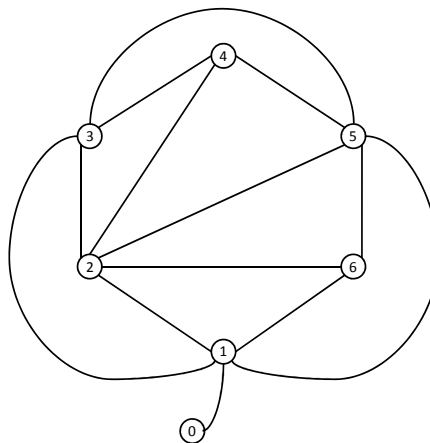


Figure 1: Example of a pyramid's map

The pyramid is described via a matrix p . $p[i]$ contains the indices of the rooms that are directly reachable from room i when considering the entrances in counter-clockwise order. For the pyramid in

Figure 1, $R = 6$ and p would write

```
p[0] = 1
p[1] = 6,2,3,0,5
p[2] = 1,6,5,4,3
p[3] = 4,5,1,2
p[4] = 3,2,5
p[5] = 3,4,2,6,1
p[6] = 5,2,1
```

Easy Task - Feasibility

Check whether there exists a way for Heidi, to enter the pyramid, fetch the calendar and get out of the pyramid again.

You are to write a function `bool fetch_calendar(vector<vector<int> > p)` that

- takes the pyramid map as a parameter. $p[i]$ has size c_i and gives room numbers connected by the corridors leaving room i , considered in counter-clockwise order.
- returns `true` if Heidi can accomplish the above mission and `false` otherwise.

For the map in Figure 1, the correct answer would be `true`.



Extension Maya

par Christian Kauth

Heidi se sent mal à l'aise concernant le calendrier à compte long Maya, gravé dans la pierre et qui se termine au treizième baktun, équivalent à notre 21 Décembre 2012. Si l'espace-temps devait se fracturer ce jour-là, non seulement il n'y aurait pas de hc² 2013, mais Heidi pourrait même courir un risque en visitant un futur lointain avec son époustouflante machine à voyager dans le temps. Ainsi, elle a décidé de rendre visite aux Mayas. Il ne fait aucun doute que la lithographie moderne trouvera assez d'espace sur la pierre du calendrier pour l'étendre à un futur au-delà de toute imagination !

Le calendrier est caché au sein d'une vaste pyramide à degrés dont Heidi possède une carte. La pyramide contient R ($1 \leq R \leq 1000$) salles cylindriques et dans chaque salle il y a jusqu'à C ($1 \leq C \leq 10$) couloirs bidirectionnels qui relient les salles entre elles. Il est à noter qu'un couloir ne ramène jamais sur la salle d'où on vient et il n'existe pas deux couloirs connectant les deux mêmes salles. L'entrée de chaque couloir est disposée d'une manière surprennament régulière autour de la circonférence de la salle et, en rentrant dans une pièce, la vue tout à fait symétrique qui attend Heidi ne l'aidera pas à s'orienter. Heureusement, une carte contient cette information. Pour chaque salle i , identifiée de 1 à R , une liste vous est donnée avec jusqu'à C destinations correspondant aux couloirs qui partent de la salle. Les destinations vous sont données dans l'ordre dans lequel sont disposés les couloirs, en faisant le tour de la salle dans le sens inverse des aiguilles d'une montre.

La liste pour la salle 3 dans la Figure 1 est (4, 5, 1, 2). Y entrer en venant de la salle 1 et prendre la deuxième sortie (toujours dans le sens inverse des aiguilles d'une montre) amènerait Heidi dans la salle 4. Pour aller à la salle 4 en venant de la salle 2, il faut choisir la première sortie. Finalement, dans ce cas, la quatrième sortie amène Heidi à revenir sur ses pas vers la salle d'où elle vient !

Nous appellerons l'extérieur de la pyramide la salle 0. Il y a exactement une entrée/sortie à la pyramide, c'est à dire qu'il n'y a qu'un seul couloir bidirectionnel connectant la salle 0 aux autres salles. Le calendrier est caché dans la salle des trésors R . Aide Heidi, qui part de l'extérieur de la pyramide (salle 0) à entrer dans la salle des trésors (salle R) où elle mettra à jour le calendrier, puis à en ressortir.

La pyramide est décrite par une matrice p . $p[i]$ contient l'indice des salles directement accessibles depuis la salle i en considérant les entrées dans le sens inverse des aiguilles d'une montre. Pour la pyramide dans la Figure 1, $R = 6$ et p serait

$p[0]$	=	1
$p[1]$	=	6, 2, 3, 0, 5
$p[2]$	=	1, 6, 5, 4, 3
$p[3]$	=	4, 5, 1, 2
$p[4]$	=	3, 2, 5
$p[5]$	=	3, 4, 2, 6, 1
$p[6]$	=	5, 2, 1

Tâche Facile - Faisabilité

Contrôlez s'il existe un chemin qui permette à Heidi d'entrer dans la pyramide, atteindre le calendrier et ressortir.

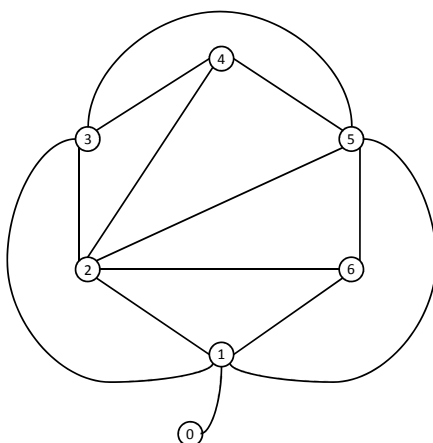


FIGURE 1 – Exemple de carte de la pyramide

Pour cela, il faut écrire une fonction `bool fetch_calendar(vector<vector<int> > p)` qui

- prenne la carte de la pyramide comme paramètre. $p[i]$ a une taille c_i et contient le numéro des salles connectées par des couloirs à la salle i , toujours dans l'ordre inverse des aiguilles d'une montre.
- retourne `true` si Heidi peut accomplir sa mission et `false` autrement.

Pour le plan de la Figure 1, la réponse correcte serait `true`.

Maya-Kalenderreform

von Christian Kauth

Seit Heidi weiss, dass der in Stein gemeisselte Mayakalender am Bak'tun 13—unserem 21. Dezember 2012—endet, fühlt sie sich unwohl. Wenn dem wirklich so wäre, dass an jenem Tag Zeit und Raum ein Ende nähmen, fände nie ein hc² 2013 statt. Und Heidi könnte ihre klasse Zeitmaschine nicht nutzen, um in eine ferne Zukunft zu reisen. . . Darum will sie den Mayas einen Besuch abstatten. Mit modernster Lithographie kann sie den Kalenderstein bestimmt für die ferne Zukunft ergänzen!

Der Kalender liegt in einer Stufenpyramide versteckt, von der Heidi eine Karte besitzt. Bestehend aus R ($1 \leq R \leq 1000$) zylindrischen Räumen, führen bis zu C ($1 \leq C \leq 10$) bidirektionale Korridore aus jedem Raum in andere Räume hinein, nie aber in den selben Raum zurück. Des weiteren verbinden keine zwei Korridore das selbe Raumpaar. Die Korridoreingänge sind erstaunlich regelmässig und sehen alle gleich aus, so dass Heidi sich ganz an der Karte orientieren muss: Für jeden Raum i , von 1 bis R durchnummeriert, enthält die Karte bis zu C Nummern. Diese bezeichnen den Zielraum der Tunnel in Raum i , geordnet nach der Reihenfolge der Tunnelleingänge (entgegen dem Uhrzeigersinn).

Die Liste für Raum 3 in Abbildung 1 ist (4, 5, 1, 2). Käme Heidi eben aus Raum 1 anspaziert, würde der zweite Ausgang entgegen dem Uhrzeigersinn sie in den Raum 4 leiten. Käme sie allerdings aus Raum 2, führte der erste Ausgang zu Raum 4. Nimmt sie den vierten Ausgang, kehrt Heidi in ihren vorherigen Raum zurück.

Nennen wir den Platz ausserhalb der Pyramide Raum 0. Die Pyramide besitzt genau einen Ein-/Ausgang, so dass nur ein Tunnel Raum 0 mit irgend einem anderen Raum verbindet. Der Kalender ist im Schatzraum R verborgen. Führt Heidi von ausserhalb der Pyramide (Raum 0) in den Schatzraum (Raum R), wo sie den Kalender verlängert, und dann wieder ins Freie.

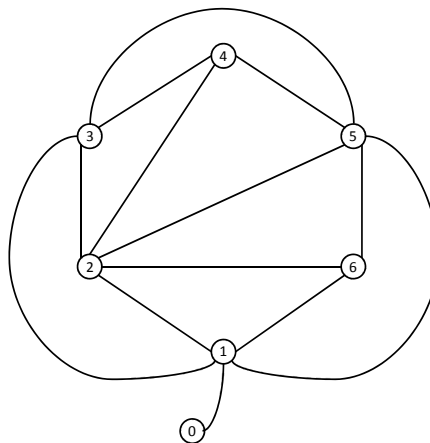


Abbildung 1: Beispiel einer Karte

Die Pyramide ist durch eine Matrix p beschrieben. $p[i]$ enthält die Nummern der Räume, welche direkt von Raum i aus erreichbar sind, aufgelistet in der Reihenfolge ihrer Tunnelleingänge entgegen

dem Uhrzeigersinn. Für die Pyramide in Abbildung 1, mit $R = 6$, schreibt sich p wie folgt

$$\begin{aligned} p[0] &= 1 \\ p[1] &= 6, 2, 3, 0, 5 \\ p[2] &= 1, 6, 5, 4, 3 \\ p[3] &= 4, 5, 1, 2 \\ p[4] &= 3, 2, 5 \\ p[5] &= 3, 4, 2, 6, 1 \\ p[6] &= 5, 2, 1 \end{aligned}$$

Einfache Aufgabe - Machbarkeit

Findet heraus ob es Heidi möglich ist in die Pyramide einzutreten, den Kalender aufzuspüren und wieder ins Freie zu gelangen.

Schreibt dazu eine Funktion `bool fetch_calendar(vector<vector<int> > p)` welche

- die Pyramide als Parameter nimmt. $p[i]$ enthält c_i Raumnummern, welche sich an den Enden der Korridore befinden, die aus Raum i rausführen, aufgelistet entgegen dem Uhrzeigersinn ihrer Tunnelleingänge.
- `true` zurückgibt falls Heidi ihre Mission erfüllen kann und `false` andernfalls.

Für die Karte der Abbildung 1 lautet die korrekte Antwort `true`.