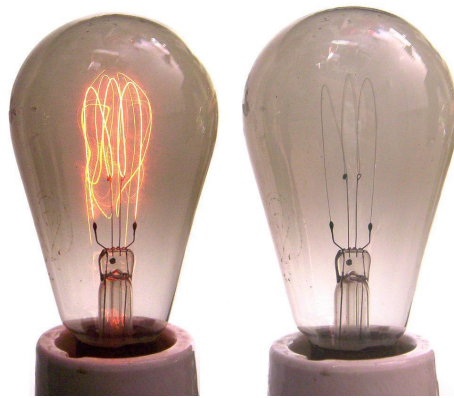


Algorithmique 2 (INFO-F203)

Fiat Lux

Jean Cardinal

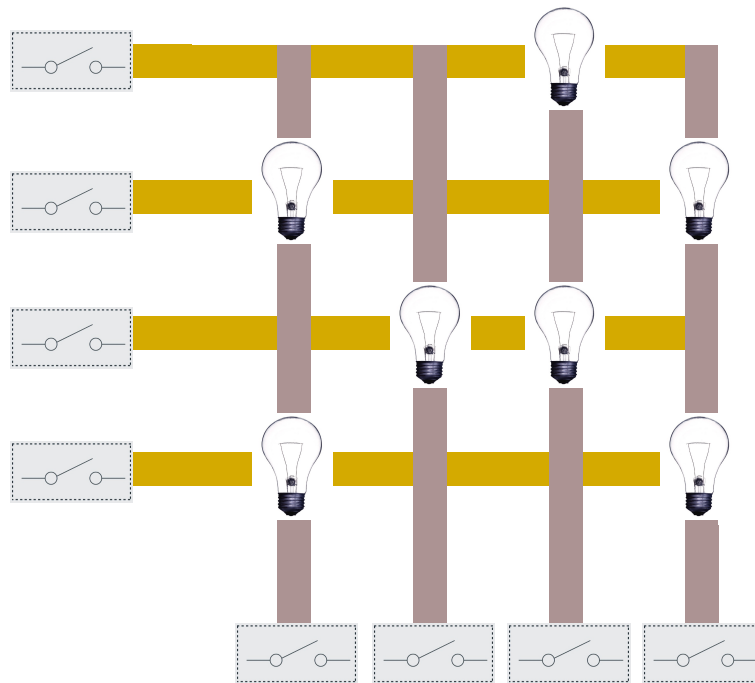
Université libre de Bruxelles (ULB)
Mars 2024



(Wikipedia – Creative Commons.)

Un problème d'éclairage

On considère une grille carrée composée de câbles horizontaux et verticaux. Des ampoules sont connectées à certaines intersections de ces câbles (pas toutes), tandis que chacune des extrémités gauches des câbles horizontaux, ainsi que chacune des extrémités inférieures des câbles verticaux est munie d'un interrupteur. On dispose donc d'un interrupteur pour chaque ligne et chaque colonne de la grille. Un exemple est donné dans la figure suivante :



Chaque interrupteur peut être en position ouverte ou fermée. Chacune des ampoules ne s'allume que pour certaines combinaisons bien définies des positions des deux interrupteurs auxquels elle est reliée, en ligne et en colonne, indépendamment des autres ampoules. À chaque ampoule sont donc associés quatre bits (non représentés sur la figure) qui indiquent, pour chacune des quatre combinaisons des positions de deux interrupteurs, si l'ampoule s'allume. Une ampoule pourrait par exemple ne s'allumer que si l'interrupteur de sa colonne est ouvert, tandis qu'une autre pourrait ne s'allumer que si ses deux interrupteurs sont dans des positions différentes.

Votre travail

Décision

Dans une première partie du travail, nous vous demandons de décrire un algorithme *efficace* pour répondre à la question suivante :

Est-il possible d'allumer toutes les ampoules simultanément ?

Il s'agit donc de décider s'il existe une configuration des interrupteurs telle que toutes les ampoules sont allumées. Nous vous demandons de :

1. décrire rigoureusement les étapes et les structures de données utilisées par votre algorithme,
2. discuter, en justifiant rigoureusement, de la complexité de l'algorithme en fonction du nombre d'ampoules et du nombre d'interrupteurs,
3. comparer la complexité de votre algorithme à une borne inférieure, et démontrer éventuellement l'optimalité de votre algorithme,
4. proposer une mise en œuvre de cet algorithme en Java.

Format d'entrée. L'instance du problème est donnée sous forme d'un fichier texte, composé d'une ligne par ampoule, et dont chaque ligne a le format suivant:

```
l c x y z t
```

où l et c sont des entiers indiquant les coordonnées en ligne et colonne de l'ampoule, et les quatre symboles x y z t sont des bits (0 ou 1) indiquant si l'ampoule s'allume pour chacune des quatre combinaisons possibles des positions des deux interrupteurs correspondant en ligne et en colonne, dans l'ordre: ouvert-ouvert, ouvert-fermé, fermé-ouvert et fermé-fermé.

Une instance du problème avec la même disposition des ampoules que sur la figure précédente est décrite par exemple par les lignes suivantes:

```
1 3 0 0 0 1
2 1 1 0 1 0
2 4 0 1 1 0
3 2 0 0 1 1
3 3 1 0 0 0
4 1 0 1 0 0
4 4 0 1 1 0
```

Indice. Pour vous mettre sur la voie d'une solution, vous pouvez considérer un graphe dirigé défini comme suit. Pour chaque interrupteur x , on construit deux nœuds $(x, \text{fermé})$ et (x, ouvert) du graphe, correspondant aux deux positions de cet interrupteur, et les arcs du graphe représentent des implications. Un arc $(u, \text{fermé}) \rightarrow (v, \text{ouvert})$, par exemple, indique que si toutes les ampoules sont allumées et u est fermé, alors v doit être ouvert.

Optimisation

Dans la seconde partie du travail, nous vous demandons de concevoir un algorithme qui permet de répondre à la question suivante :

Quel est le plus grand nombre d'ampoules que l'on peut allumer simultanément ?

Le problème est plus général que dans la première partie : l’algorithme de décision répond positivement si et seulement si le maximum est égal au nombre total d’ampoules. Comme dans la première partie, nous vous demandons une description et une analyse détaillée de votre proposition d’algorithme, ainsi qu’une mise en œuvre en Java. Le format d’entrée des instances est le même.

Consignes

- Votre travail sera composé d’un rapport scientifique, d’un programme en Java, et d’un fichier README.
- Le fichier README est un fichier texte devant contenir les instructions permettant au correcteur de compiler, exécuter et tester le code.
- Le rapport devra être au format pdf, rédigé en français correct, et mis en page avec le logiciel \LaTeX .
- Le rapport comprendra également la liste des références bibliographiques utilisées.
- On s’attend pour ce travail à un rapport dont la longueur est d’environ une dizaine de pages.
- Les fichiers sont à remettre sur l’Université Virtuelle (UV).

Recommandations et ressources externes

Rédaction scientifique. Des conseils généraux sur la rédaction d’un rapport scientifique se trouvent dans le document “Éléments de rédaction scientifique en informatique”, rédigé par Hadrien Mélot (UMons) et disponible à l’adresse suivante :

<http://informatique.umons.ac.be/algo/redacSci.pdf>.

Ressources bibliographiques. Vous êtes encouragé·e·s, pour répondre aux questions, à consulter la littérature scientifique. Les documents utilisés doivent être de nature académique ou didactique, clairement mentionnés dans le texte et figurer dans la liste de références bibliographiques. Nous vous encourageons à utiliser le logiciel BibTeX :

<https://www.bibtex.com/>.

Code. Les programmes doivent être dûment commentés et structurés, et respecter les bonnes pratiques enseignées dans les cours de programmation orientée objet. Vous êtes autorisé·e·s à réutiliser, en le mentionnant clairement, les classes de la bibliothèque Java `algs4.jar`, disponible à l'adresse suivante :

<https://algs4.cs.princeton.edu/code/>.

Travail en binôme, échanges et plagiat

Vous êtes autorisé·e·s à rendre le travail en collaboration avec un·e autre étudiant·e, auquel cas la même note sera attribuée aux deux membres du binôme. Vous pouvez également rendre le travail à titre individuel.

Vous êtes autorisé·e·s à discuter des problèmes et de vos solutions éventuelles avec d'autres étudiant·e·s et d'autres binômes. Si un point de votre rapport est le fruit de ces discussions, vous êtes invité·e·s à le mentionner explicitement. La rédaction du rapport et des programmes doit cependant être propre à chaque étudiant·e / binôme. Tout emprunt non mentionné explicitement est un plagiat, et constitue une fraude. Les étudiant·e·s reconnu·e·s coupables de fraude ou de tentative de fraude s'exposent à des sanctions disciplinaires comprenant l'annulation de la session d'examens et l'interdiction de s'inscrire à la session d'examens suivante.

<https://bib.ulb.be/fr/support/boite-a-outils/evitez-le-plagiat>

Date de remise

29 avril 2024, avant minuit.