



INFORMATIK-BIBER SCHWEIZ CASTOR INFORMATIQUE SUISSE CASTORO INFORMATICO SVIZZERA

Exercices 2025

Années HarmoS 13/14/15



<https://www.castor-informatique.ch/>

Éditeurs:

Susanne Thut, Nora A. Escherle,
Elsa Pellet, Jean-Philippe Pellet

010100110101011001001001
010000010010110101010011
010100110100100101000101
001011010101001101010011
010010010100100100100001

SS! E

www.svia-ssie-ssii.ch
schweizerischervereinfürinformatikind
erausbildung//sociétésuissespourl'infor
matique dans l'enseignement//societàsviz
zeraperl'informaticanell'insegnamento





Ont collaboré au Castor Informatique 2025

Masiar Babazadeh, Jean-Philippe Pellet, Andrea Maria Schmid, Giovanni Serafini, Susanne Thut

Cheffe de projet : Nora A. Escherle

Nous adressons nos remerciements pour le travail de développement des exercices du concours à:
Patricia Heckendorf, Gymnasium Kirschgarten

Juraj Hromkovič, Regula Lacher: ETH Zürich, Ausbildungs- und Beratungszentrum für Informatikunterricht

Jens Hartmann, Stephan Koch, Dirk Schmerenbeck und Jacqueline Staub: Universität Trier, Allemagne

Le choix des exercices a été fait en collaboration avec les organisateurs de Bebras en Allemagne, Autriche et Hongrie. Nous remercions en particulier :

Philip Whittington, Silvan Horvath: ETH Zürich, Ausbildungs- und Beratungszentrum für Informatikunterricht

Wolfgang Pohl, Karsten Schulz, Franziska Kaltenberger, Margaretha Schlüter, Kirsten Schlüter, Michael Weigend: Bundesweite Informatikwettbewerbe (BWINF), Allemagne

Wilfried Baumann: Österreichische Computer Gesellschaft, Autriche

Gerald Futschek, Lukas Lehner: Technische Universität Wien, Autriche

Zsuzsa Pluhár, Bence Gaal: ELTE Informatikai Kar, Hongrie

La version en ligne du concours a été réalisée sur l'infrastructure cuttle.org. Nous remercions pour la bonne collaboration :

Eljakim Schrijvers, Justina Oostendorp, Alieke Stijf, Kyra Willekes: cuttle.org, Pays-Bas

Andrew Csizmadia: Raspberry Pi Foundation, Royaume-Uni

Les tâches de programmation ont été créées et développées spécialement pour la plate-forme en ligne. Nous remercions chaleureusement pour leur initiative :

Jacqueline Staub: Universität Trier, Allemagne

Dirk Schmerenbeck: Universität Trier, Allemagne

Dave Oostendorp: cuttle.org, Pays-Bas

Pour le support pendant les semaines du concours, nous remercions en plus :

Eveline Moor: Société suisse pour l'informatique dans l'enseignement

Nous remercions les personnes suivantes pour l'organisation et la réalisation de la finale suisse :

Dennis Komm, Hans-Joachim Böckenhauer, Angélica Herrera Loyo, Andre Macejko, Moritz Stocker, Philip Whittington, Silvan Horvath: ETH Zürich, Ausbildungs- und Beratungszentrum für Informatikunterricht

Pour la correction des épreuves :

Clemens Bachmann, Morel Blaise, Tobias Boschung, Davud Evren, Jay Forrer, Sven Grübel, Urs Hauser, Fabian Heller, Jolanda Hofer, Alessandra Iacopino, Saskia Koller, Richard Královič, Jan Mantsch, Adeline Pittet, Alexander Skodinis, Emanuel Skodinis, Jasmin Sudar, Valerie Verdan, Chris



Wernke

Pour la traduction française des épreuves :

Jean-Philippe Pellet : Haute école pédagogique du canton de Vaud

Christoph Frei : Chragokyberneticks (Logo Castor Informatique Suisse)

Andrea Leu, Sarah Beyeler, Maggie Winter : Senarcens Leu + Partner AG

Des remerciements particuliers sont dûs pour leur grand soutien à Juraj Hromkovič, Dennis Komm, Gabriel Parriaux et la Fondation Hasler. Sans eux, ce concours n'existerait pas.

La version allemande des exercices a également été utilisée en Allemagne et en Autriche.

L'adaptation française a été réalisée par Elsa Pellet et l'adaptation italienne par Christian Giang.



**INFORMATIK-BIBER SCHWEIZ
CASTOR INFORMATIQUE SUISSE
CASTORO INFORMATICO SVIZZERA**

Le Castor Informatique 2025 a été réalisé par la Société suisse pour l'informatique dans l'enseignement (SSIE) et généreusement soutenu par la Fondation Hasler. Parmi les autres partenaires et sponsors qui ont soutenu financièrement le concours, citons Abraxas Informatik AG, l'Office de l'école obligatoire et du conseil (OEKO) du canton de Berne, l'Office de l'économie (AWI) du canton de Zurich, CYON AG et UBS.

Cette brochure a été produite le 10 décembre 2025 avec le système de composition de documents L^AT_EX. Nous remercions Christian Datzko pour le développement et maintien de la structure de génération des 36 versions de cette brochure (selon les langues et les degrés). La structure actuelle a été mise en place de manière similaire à la structure précédente, qui a été développée conjointement avec Ivo Blöchliger dès 2014. Nous remercions aussi Jean-Philippe Pellet pour le développement de la série d'outils **bebras**, qui est utilisée depuis 2020 pour la conversion des documents source depuis les formats Markdown et YAML.

Tous les liens dans les tâches ci-après ont été vérifiés le 1^{er} décembre 2025.



Les exercices sont protégés par une licence Creative Commons Paternité – Pas d'Utilisation Commerciale – Partage dans les Mêmes Conditions 4.0 International. Les auteur·e·s sont cité·e·s en p. 22.



Préambule

Très bien établi dans différents pays européens et plus largement à l'échelle mondiale depuis plusieurs années, le concours «Castor Informatique» a pour but d'éveiller l'intérêt des enfants et des jeunes pour l'informatique. En Suisse, le concours est organisé en allemand, en français et en italien par la SSIE, la Société suisse pour l'informatique dans l'enseignement, et soutenu par la Fondation Hasler.

Le Castor Informatique est le partenaire suisse du concours «Bebras International Contest on Informatics and Computer Fluency» (<https://www.bebras.org/>), initié en Lituanie.

Le concours a été organisé pour la première fois en Suisse en 2010. Le Petit Castor (années HarmoS 5 et 6) a été organisé pour la première fois en 2012.

Le Castor Informatique vise à motiver les élèves à apprendre l'informatique. Il souhaite lever les réticences et susciter l'intérêt quant à l'enseignement de l'informatique à l'école. Le concours ne suppose aucun prérequis quant à l'utilisation des ordinateurs, sauf de savoir naviguer sur Internet, car le concours s'effectue en ligne. Pour répondre, il faut structurer sa pensée, faire preuve de logique mais aussi d'imagination. Les exercices sont expressément conçus pour développer un intérêt durable pour l'informatique, au-delà de la durée du concours.

Le concours Castor Informatique 2025 a été fait pour cinq tranches d'âge, basées sur ces années scolaires :

- Années HarmoS 5 et 6
- Années HarmoS 7 et 8
- Années HarmoS 9 et 10
- Années HarmoS 11 et 12
- Années HarmoS 13 à 15

Chaque tranche d'âge avait des exercices classés en trois niveaux de difficulté : facile, moyen et difficile. Les élèves des années HarmoS 5 et 6 avaient 9 exercices à résoudre : 3 faciles, 3 moyens, 3 difficiles. Les élèves des années HarmoS 7 et 8 avaient, quant à eux, 12 exercices à résoudre (4 de chaque niveau de difficulté). Finalement, chaque autre tranche d'âge devait résoudre 15 exercices (5 de chaque niveau de difficulté).

Chaque réponse correcte donnait des points, chaque réponse fausse réduisait le total des points. Ne pas répondre à une question n'avait aucune incidence sur le nombre de points. Le nombre de points de chaque exercice était fixé en fonction du degré de difficulté :

	Facile	Moyen	Difficile
Réponse correcte	6 points	9 points	12 points
Réponse fausse	-2 points	-3 points	-4 points

Utilisé au niveau international, ce système de distribution des points est conçu pour limiter le succès en cas de réponses données au hasard.



Chaque participant·e obtenait initialement 45 points (ou 27 pour les années HarmoS 5 et 6, et 36 pour les années HarmoS 7 et 8).

Le nombre de points maximal était ainsi de 180 (ou 108 pour les années HarmoS 5 et 6, et 144 pour les années HarmoS 7 et 8). Le nombre de points minimal était zéro.

Les réponses de nombreux exercices étaient affichées dans un ordre établi au hasard. Certains exercices ont été traités par plusieurs tranches d'âge (en étant classés différemment dans les niveaux de difficulté).

Certains exercices sont indiqués comme « bonus » pour certaines catégories d'âge : ils ne comptent pas dans le total des points, mais servent à départager plusieurs scores identiques en cas de qualification pour les éventuels tours suivants.

Pour de plus amples informations :

Société suisse pour l'informatique dans l'enseignement

SVIA-SSIE-SSII

Castor Informatique

Jean-Philippe Pellet

<https://www.castor-informatique.ch/kontaktieren/>

<https://www.castor-informatique.ch/>



Table des matières

Ont collaboré au Castor Informatique 2025	i
Préambule	iii
Table des matières	v
1. Message secret	1
2. Cerf-volant perdu	2
3. Couronne de l'Avent	3
4. Filtre d'image	4
5. Transport de farine	5
6. Noir et blanc	6
7. Adresses e-mail	7
8. Acide castorique	9
9. Transports publics	10
10. Visite de Séoul	11
11. Lacs de montagne	12
12. Parking	13
13. Castor Jones	14
14. Session d'examens	16
15. Castor de contrôle	17
16. Papier, caillou, ciseaux	18
17. D'un côté à l'autre	21
A. Auteur-e-s des exercices	22
B. Partenaires académiques	24
C. Sponsoring	25



1. Message secret

Une image secrète composée de pixels noirs et blancs doit être transmise de manière sûre. Pour cela, le service de messagerie crée deux images composées de pixels foncés et clairs sur des feuilles transparentes. L'image secrète n'est révélée que lorsque les deux feuilles transparentes sont superposées.

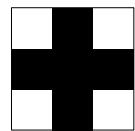
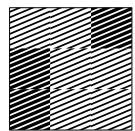
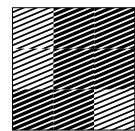


Image secrète



Transparent 1



Transparent 2

Transparents 1 et 2
superposés

Les images des les feuilles transparentes sont créées de la manière suivante: tout d'abord, un motif aléatoire de pixels clairs et foncés est imprimé sur la première feuille transparente. La couleur des pixels de la deuxième feuille transparente est déterminée par la règle suivante en fonction de la couleur des mêmes pixels sur l'image originale et sur la première feuille transparente:

- Si le pixel de l'image originale est noir , les pixels sur le transparent 1 et le transparent 2 doivent être de couleurs différentes (l'un clair et l'autre foncé).
- Si le pixel de l'image originale est blanc , les pixels sur le transparent 1 et le transparent 2 doivent être la même couleur (les deux clairs ou les deux foncés).

Le premier transparent a déjà été imprimé pour l'image ci-dessous. Détermine la couleur des pixels du deuxième transparent.

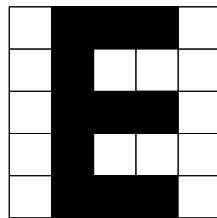
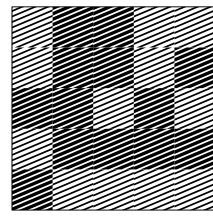
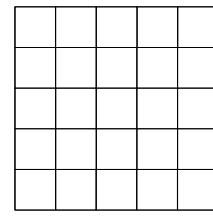


Image secrète



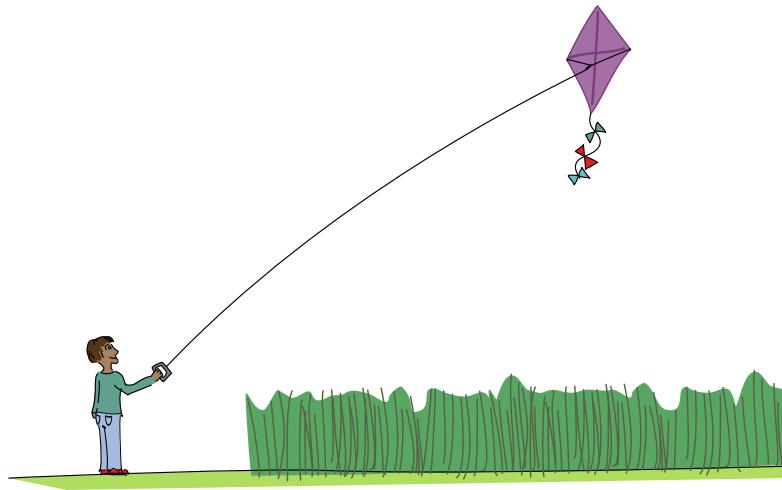
Transparent 1



Transparent 2



2. Cerf-volant perdu

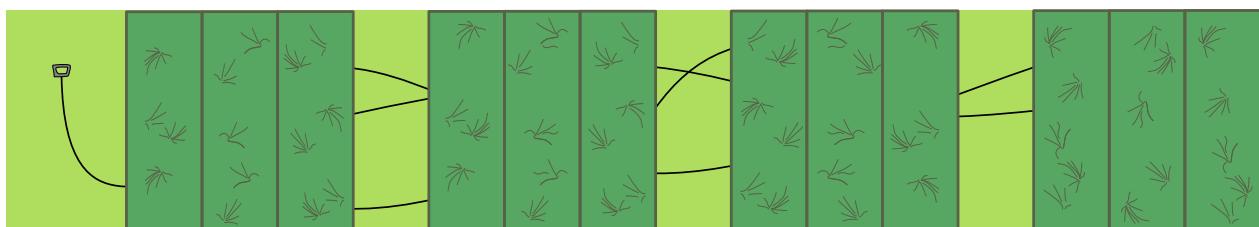


Pas de chance! Asterios a perdu son cerf-volant dans la prairie. Le fil s'est emmêlé dans les hautes herbes et c'est difficile de retrouver le cerf-volant.

La prairie est divisée en 15 cases qui peuvent être examinées une à une.

Asterios a déjà examiné 3 cases de la prairie. Il regarde attentivement comment le fil est arrangé dans ces cases et remarque qu'il ne doit plus examiner qu'une seule case pour savoir exactement dans quelle case est son cerf-volant.

De quelle case s'agit-il?





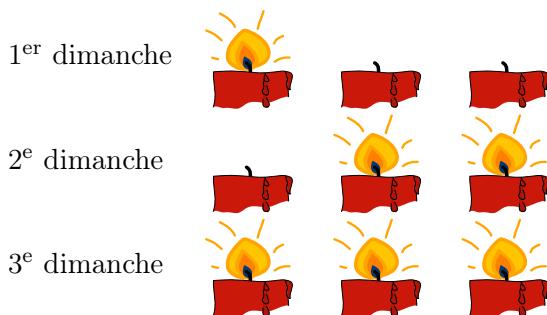
3. Couronne de l'Avent

Il existe une tradition qui consiste à allumer des bougies lors des quatre dimanches de l'Avent : une bougie le premier dimanche, deux bougies le deuxième dimanche, et ainsi de suite.

Chris adore cette tradition. Ses quatre bougies ont la même taille avant d'avoir été allumées. Pour passer un Noël magnifique, Chris aimerait que ses bougies aient aussi toutes la même taille après le dernier dimanche de l'Avent. Pour cela, il devrait allumer chaque bougie le même nombre de fois.

Malheureusement, Chris ne trouve pas comment passer un Noël magnifique.

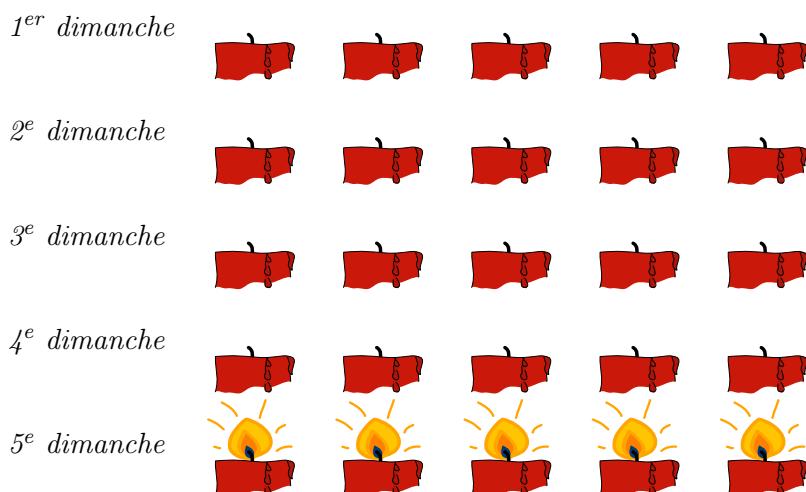
Si la tradition ne concernait que 3 dimanches (et 3 bougies), ce serait possible : Chris allumerait chaque bougie deux fois.



Ce serait aussi possible avec 5 dimanches (et 5 bougies).

Montre à Chris comment allumer chaque bougie le même nombre de fois.

Nous avons déjà allumé les bougies pour le cinquième dimanche.



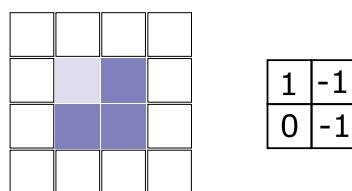


4. Filtre d'image

Les images numériques sont souvent composées de pixels. Sandra crée des *cartes de teinte* pour de telles images pixelisées. Pour cela, elle commence par ajouter aux images un cadre de pixels blancs. Ensuite, elle détermine une valeur de teinte pour chaque pixel de l'image comme suit:

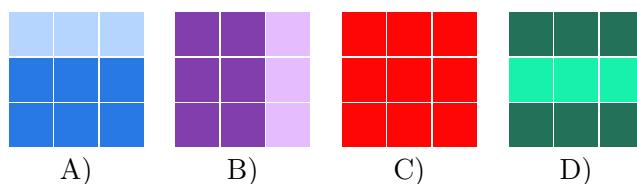
1	■	1 si le pixel est plus clair que son voisin de droite;
0	■	0 si le pixel a la même teinte que son voisin de droite;
-1	■	-1 si le pixel est plus foncé que son voisin de droite.

Voici une image composée de 4 pixels (plus le cadre de pixels blancs) et la carte de teinte correspondante.



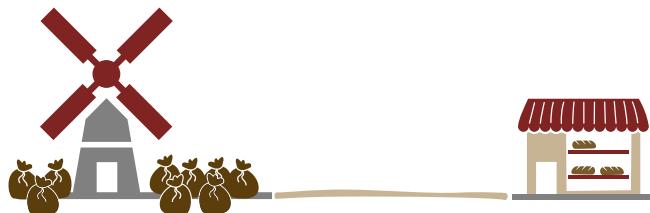
Tu vois ci-dessous 4 images de 9 pixels chacune. Trois d'entre elles ont la même carte de teinte.

Quelle est l'image qui a une carte de teinte différente ?





5. Transport de farine



Albert et Marco travaillent dans une boulangerie. Ils doivent souvent aller chercher de la farine au moulin. Ils ne peuvent pas y aller en même temps, car l'un d'entre eux doit servir les clients dans la boulangerie.

Ils ne transportent pas la même quantité de farine à la même vitesse:



Albert transporte 13 kg de farine en 1 heure.

Marco transporte 5 kg de farine en 30 minutes.

Chacun doit se reposer pendant 30 minutes après 3 transports de farine. Ils peuvent servir des clients en se reposant.

Albert et Marco aimeraient transporter le plus de farine possible en 8 heures. C'est possible dans une seule des conditions suivantes. Laquelle ?

- A) Albert doit y aller en premier.
- B) Marco doit y aller en premier.
- C) Marco doit y aller en dernier.
- D) Albert ne doit pas y aller en dernier.
- E) Marco doit y aller une seule fois.



6. Noir et blanc

Sarah aimerait représenter des suites de carrés noirs et blancs avec des lettres. Pour cela, elle applique l'algorithme suivant à une suite de lettres :

- Si tous les carrés de la suite sont blancs, écris B.
- Si tous les carrés de la suite sont noirs, écris N.
- Si la suite contient des carrés noirs et des carrés blancs, écris x et :
 - applique l'algorithme à la moitié gauche de la suite, puis
 - applique l'algorithme à la moitié droite de la suite.

Voici les représentations données par l'algorithme pour quelques suites de carrés :

	B
	xBN
	xxNBN
	xNxBxBNB

Quelle est la représentation donnée par l'algorithme de Sarah pour la suite de carrés suivante ?





7. Adresses e-mail

Les castors du service informatique ont besoin d'un système qui reconnaît si une suite de caractères est une adresse e-mail castor. Une adresse e-mail castor est faite de trois éléments:



	Nom	Explication	Valeurs possibles
(1)	Identifier	N'importe quelle suite de caractères non vide avec des lettres minuscules et/ou des chiffres	0–9, a–z
(2)	Arobase		@
(3)	Nom du serveur		
(3a)	Nom de domaine	N'importe quelle suite de caractères non vide avec des lettres minuscules et/ou des chiffres	0–9, a–z
(3b)	Point		.
(3c)	Nom de domaine de premier niveau	N'importe quelle suite de caractères non vide avec des lettres minuscules et/ou des chiffres	0–9, a–z

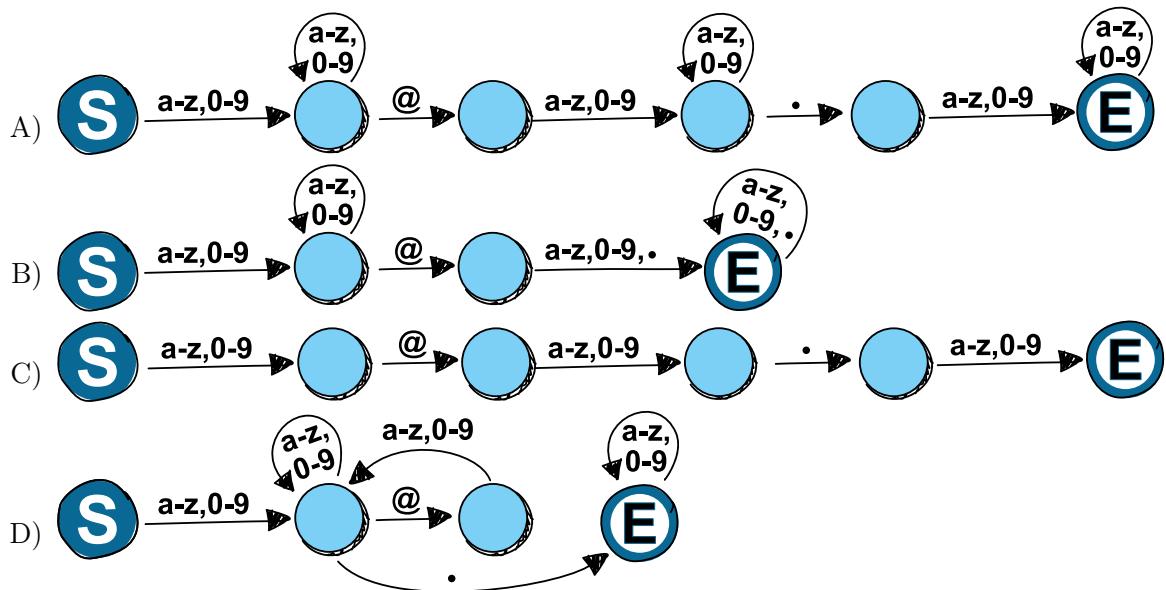
Une suite «non vide» est composée d'au moins un caractère. «bebra@sapin2.enaval» est un exemple d'adresse e-mail castor.

Les castors du service informatique développent quatre systèmes et les représentent à l'aide de diagrammes composés de cercles et de flèches. Chaque cercle représente un état dans lequel le système peut être. Chaque flèche représente le changement à l'état suivant, état qui peut être le même que le précédent. L'annotation de la flèche nous dit à quel caractère ce changement d'état correspond.

Chaque système examine une suite de caractères un caractère après l'autre, de gauche à droite et se trouve dans l'état **S** au départ. L'état suivant dépend du caractère actuellement examiné: le système suit la flèche de changement qui correspond au caractère courant. Si le système se trouve dans l'état **E** à la fin de la suite de caractère, celle-ci a été reconnue comme adresse e-mail castor.



Seul un de ces systèmes identifie correctement toutes les adresses e-mail castor. Lequel ?





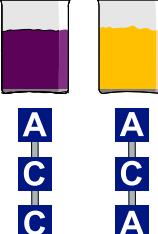
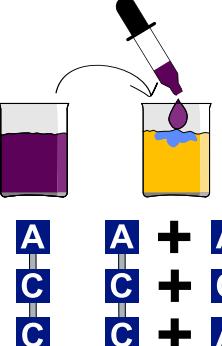
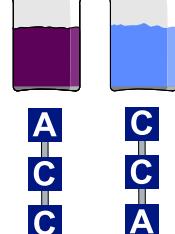
8. Acide castorique

Les acides castoriques sont des substances chimiques spéciales : leurs molécules sont faites de suites de trois éléments qui peuvent chacun être de type A ou C. Cette suite est utilisée pour nommer la molécule d'acide castorique, par exemple ACA.

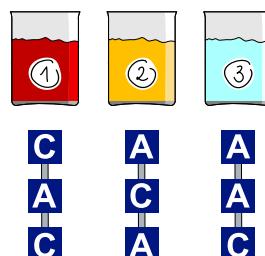
Lorsqu'un acide castorique est ajouté à un autre, un nouvel acide castorique se forme. Les éléments des deux acides de départ forment le nouvel acide d'après les règles suivantes :



Voici un exemple : si on ajoute une goutte d'ACC à de l'ACA, on obtient du CCA :

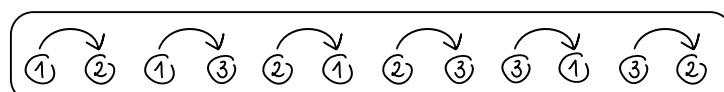
Acides ACC et ACA	Goutte d'ACC dans ACA	CCA se forme à droite
		

Au laboratoire, il y a trois récipients, 1, 2 et 3, contenant les acides castoriques CAC, ACA et AAC :



Une telle instruction te permet spécifier qu'une goutte de l'acide castorique d'un récipient (par exemple le récipient 2) doit être ajoutée à l'acide castorique d'un autre récipient (par exemple le récipient 3).

Il y a 6 instructions ci-dessous. Utilises-en le moins possible pour échanger les acides castoriques des récipients 1 et 3 : à la fin, AAC doit être dans le récipient 1, ACA rester dans le récipient 2 et CAC être dans le récipient 3.





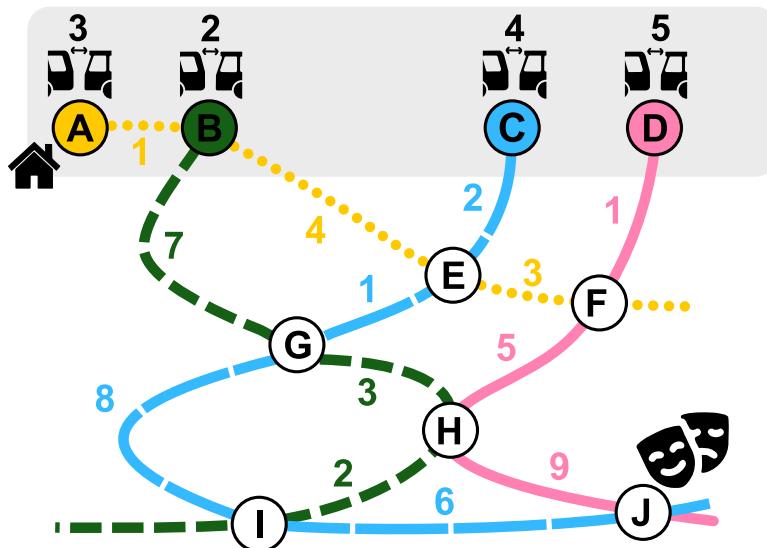
9. Transports publics

Il y a 4 lignes de bus dans la ville de Marcus. Elles partent des arrêts A, B, C et D, respectivement. Tu peux voir leur itinéraire sur le plan ci-dessous.

Les premiers bus de chaque ligne partent tous au même moment des arrêts de départ (A, B, C, D): c'est la minute 0. Ensuite, les bus partent aux intervalles indiqués . Par exemple, un bus part toutes les 3 minutes de l'arrêt A, donc aux minutes 0, 3, 6, 9, etc.

Chaque tronçon d'itinéraire entre deux arrêts indique le nombre de minutes nécessaire à parcourir le tronçon. Par exemple, le premier bus partant de l'arrêt A atteint l'arrêt F à la minute $0 + 1 + 4 + 3 = 8$.

Marcus habite la maison près de l'arrêt A. Depuis là, il aimeraient prendre le premier bus pour aller jusqu'au théâtre . Il peut changer de bus en 0 minute aux arrêts auxquels deux lignes se croisent. Il peut donc continuer sa route avec n'importe quel bus qui arrive en même temps ou plus tard que lui au même arrêt. Marcus sait quelles lignes prendre et où changer pour arriver au théâtre le plus tôt possible.

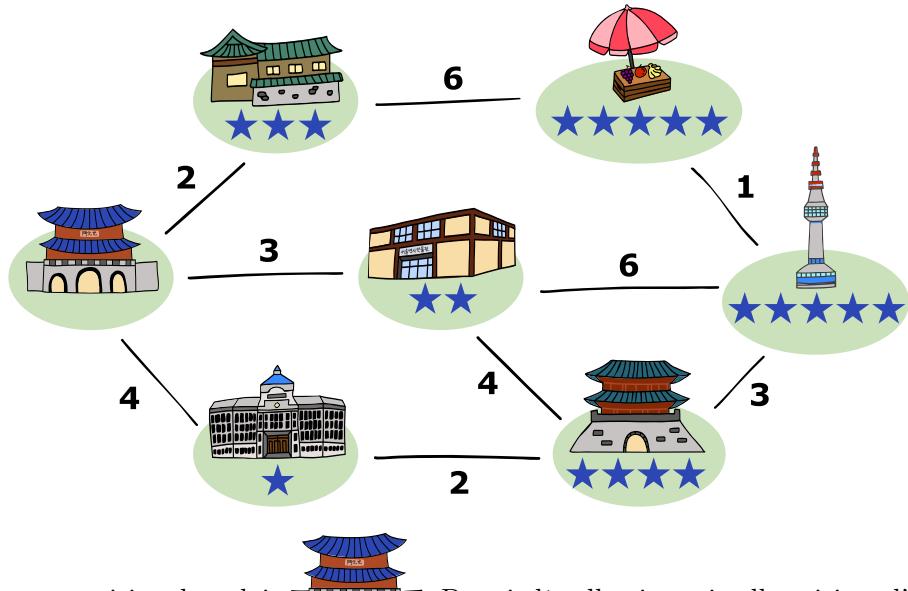


Quels arrêts se trouvent sur la route de Marcus ?



10. Visite de Séoul

À Séoul, en Corée, il y a des bus qui relient les sites importants pour les touristes. L'image montre les sites les plus importants de Séoul. Plus un site est apprécié, plus il a d'étoiles. Les lignes montrent les connexions par bus. La longueur de chaque ligne en kilomètres est notée à côté de la ligne.



Lotte commence par visiter le palais  . Depuis là, elle aimerait aller visiter d'autres sites en bus. Elle a un billet avec lequel elle peut rouler au maximum 10 kilomètres. Elle aimerait l'utiliser pour visiter des sites avec le plus grand nombre d'étoiles possible au total. Elle ne visite un endroit qu'une seule fois et ne doit pas revenir au palais.

Quelles connexions Lotte doit-elle emprunter pour récolter le plus grand nombre d'étoiles ?



11. Lacs de montagne

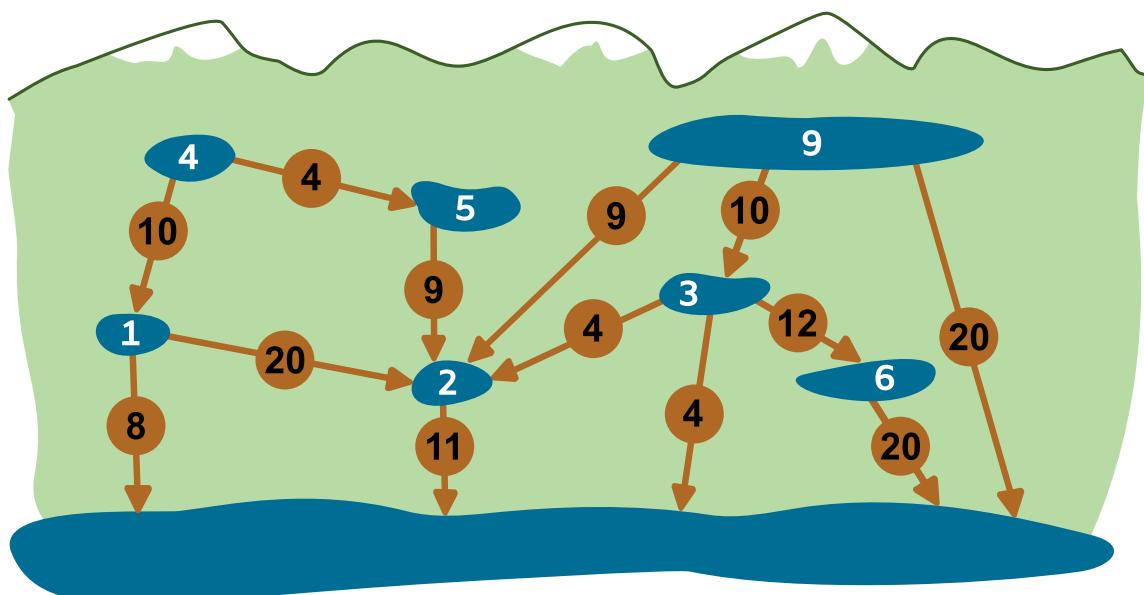
Dans la montagne au-dessus du réservoir du barrage, il y a plusieurs petits lacs naturels. Ils pourraient déborder s'il pleut beaucoup, ce qui est dangereux. C'est pour cela que des canaux doivent être construits entre certains lacs. Ces canaux doivent amener toute l'eau des lacs en trop dans le réservoir du barrage. Leur construction doit coûter le moins cher possible.

Chaque lac de montagne a un nombre qui indique combien d'eau en surplus doit en être déviée.

Une flèche dénote chaque endroit auquel un canal peut être construit. Elle montre la direction de l'eau, et le nombre indique la capacité du canal (donc combien d'eau il peut transporter) ainsi que son prix.

Fais attention : lorsqu'un canal transporte l'eau d'un lac de montagne à un autre, l'eau en trop des deux lacs s'accumule dans le deuxième lac.

Où faut-il construire des canaux ?





12. Parking

Il y a 9 invités qui viennent en voiture à une fête. Devant la salle de fête, 9 voitures peuvent se parquer dans 3 colonnes avec 3 voitures les unes derrière les autres dans chaque colonne. Les invités arrivent dans l'ordre suivant :

Anja, Brigitte, Clara, David, Elia, Frank, Gabi, Hugo et enfin Julia.

Pour se parquer, chaque invité choisit une colonne et s'y parque le plus en avant possible.

Les invités veulent partir de la fête dans l'ordre suivant :

Gabi, David, Brigitte, Elia, Julia, Clara, Hugo, Anja et enfin Frank.

Les voitures d'**A**nja, **B**rigitte et **C**larla sont déjà parquées. Les autres invités se parquent les uns après les autres. Ils veulent se parquer de manière à ce que personne ne soit bloqué par une autre voiture au moment de partir.



Montre aux invités comment ils peuvent se parquer!

Place les 6 voitures restantes dans les colonnes du parking. Tu dois considérer l'ordre d'arrivée et l'ordre de départ.



13. Castor Jones



Castor Jones est dans une terrible pyramide remplie de couloirs dangereux. Au bout de chaque couloir se trouve un fabuleux trésor. Jones veut atteindre chaque trésor *le plus vite possible*.

Chaque couloir est bloqué par une série de blocs de pierre. Au départ, tous les blocs sont en bas. Dès que quelqu'un entre dans le couloir, ils commencent à se déplacer. Chaque bloc se déplace de haut en bas à un rythme régulier. Par exemple, un bloc avec le tempo 2 va monter d'un coup au bout de deux minutes, puis redescendre tout aussi vite deux minutes plus tard.

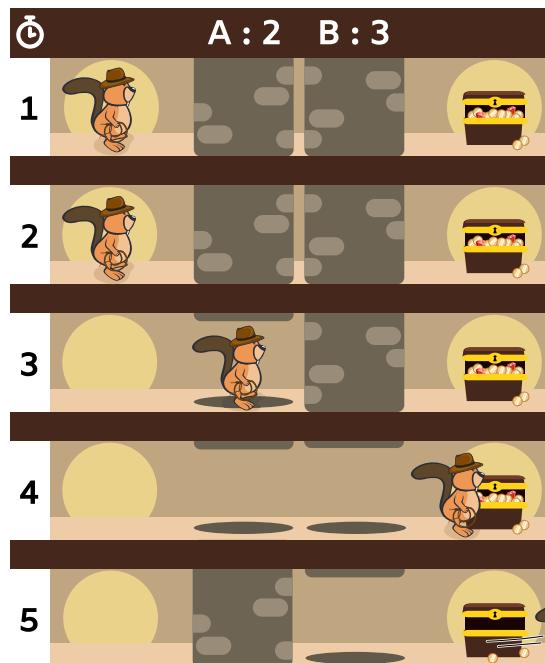
Jones se retrouve devant sa première épreuve:



Ce couloir a deux blocs: le bloc A a le tempo 2 et le bloc B le tempo 3. Heureusement, Jones a trouvé une note avec des instructions permettant d'atteindre le trésor le plus vite possible:

```
wait(2)  
goto_block(A)  
wait(1)  
goto_treasure
```

Jones suit les instructions: il attend 2 minutes, puis va vers le bloc A, y attend 1 minute et va ensuite vers le trésor. Il atteint le trésor après 3 minutes:





Jones se rend compte qu'il aurait pu atteindre le trésor à la même vitesse avec moins d'instructions:

```
wait(3)  
goto_treasure
```

Il arrive au couloir suivant. Celui-ci a quatre blocs qui ont les tempos 3, 5, 8 et 4.

Quelle est la plus courte séquence d'instructions permettant à Jones d'atteindre le trésor le plus vite possible ?





14. Session d'examens

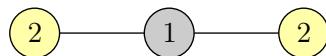
C'est la saison des examens de maturité au gymnase Euler. La session d'examens doit durer au maximum 5 jours. Comme on ne peut passer qu'un examen par jour, 2 examens qu'un même élève doit passer ne peuvent pas avoir lieu le même jour. Ces 2 examens sont alors en «conflict».

Pour faciliter la planification, les conflits sont représentés dans un diagramme fait de cercles et de lignes :

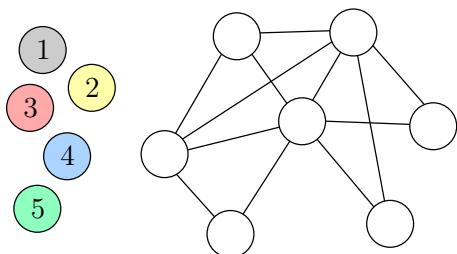
- Un cercle vide est dessiné pour chaque examen ;
- Une ligne est dessinée entre deux examens si (et seulement si) ces 2 examens sont en conflit et ne peuvent pas être prévus pour le même jour.

On écrit ensuite dans les cercles le numéro du jour (1 à 5) auquel l'examen se passera.

Voici un exemple avec 3 examens : l'examen du milieu se passe le premier jour et a 2 conflits, un avec chacun des deux autres examens, qui, eux, se passent le deuxième jour et n'ont pas de conflit entre eux.



Dans les 5 prochains jours, 7 examens doivent être prévus. Le diagramme suivant montre leurs conflits :



Répartis les examens sur le moins de jours possible (1 à 5) en tenant compte des conflits. Il y a plusieurs bonnes réponses.



15. Castor de contrôle

Le chef castor engage 4 castors *messagers*: ils envoient des messages en levant des drapeaux. Chaque castor messager peut lever soit un drapeau jaune , soit un drapeau rouge .

Il arrive qu'un castor messager se trompe et lève le mauvais drapeau. Le chef castor aimerait pouvoir détecter ces erreurs. Pour cela, il engage en plus trois castors *contrôleurs*.

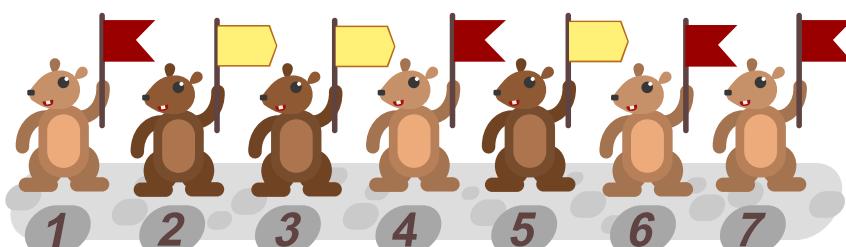
Chaque castor contrôleur vérifie les drapeaux de trois castors messagers. Si ces trois castors doivent lever un nombre impair de drapeaux rouges, le castor contrôleur doit lui aussi lever un drapeau rouge, et un drapeau jaune sinon. Lorsque les bons drapeaux sont levés, il y a ainsi un nombre pair de drapeaux rouges entre le castor contrôleur et ses 3 messagers.

Le chef numérote les castors messagers et contrôleurs et les groupe comme cela:

Castors messagers	Castor contrôleur
1, 2, 3	5
1, 2, 4	6
2, 3, 4	7

Maintenant, 7 drapeaux sont maintenant levés pour chaque message. Le chef castor voit le message ci-dessous. Il remarque qu'exactement un castor lève le mauvais drapeau.

Quel castor lève le mauvais drapeau ?

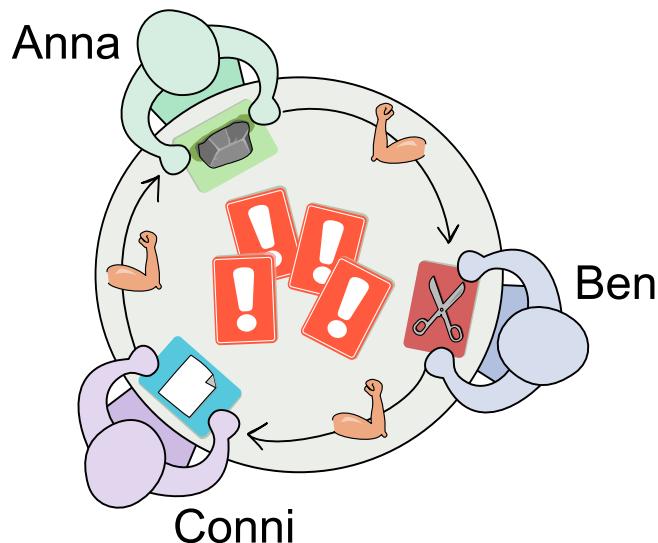




16. Papier, caillou, ciseaux

Anna, Ben et Conní jouent ensemble à «Papier, caillou, ciseaux». Ils jouent les uns contre les autres par groupes de 2, mais tous en même temps.

Chacun a tiré une carte: Anna, la carte caillou ; Ben, la carte ciseaux ; et Conní, la carte papier :



Ils jouent d'après les règles classiques: le caillou bat les ciseaux, les ciseaux battent le papier, et le papier bat le caillou. Avec la distribution actuelle des cartes, chacun perd une fois et gagne une fois. Ben, par exemple, gagne contre Conní et perd contre Anna.

Avant de déterminer le résultat du tour de jeu pour chaque joueur, il faut encore choisir l'une des quatre cartes d'action et l'appliquer. Chacune des cartes décrit un ou plusieurs échanges de cartes à faire entre 2 joueurs. Si rien d'autre n'est précisé sur la carte, les joueurs qui échangent les cartes peuvent changer à chaque échange.

Ben veut absolument gagner contre Conní, quelle que soit la manière dont l'échange sur la carte est effectué.

Une seule des 4 actions garantit que Ben gagne contre Conní. Laquelle ?

- A) Ben et Conní échangent leurs cartes un nombre impair de fois.
- B) N'importe quelle paire de joueurs sauf Ben et Conní échangent leurs cartes aussi souvent qu'ils le veulent.
- C) N'importe quelle paire de joueurs, mais au moins une fois Ben et Conní, échangent leurs cartes aussi souvent qu'ils le veulent.
- D) N'importe quelle paire de joueurs échangent leurs cartes un nombre pair de fois.



Tâches de programmation

Les tâches qui suivent sont des tâches de programmation et font partie des tâches bonus du concours.

Alors que les tâches de base n'ont aucun prérequis en informatique, ces tâches-ci se résolvent plus facilement en ayant des connaissances en programmation.

Comme la programmation sur papier n'est pas très pratique, un code QR est fourni pour chaque tâche pour la résoudre en ligne de façon interactive.





17. D'un côté à l'autre

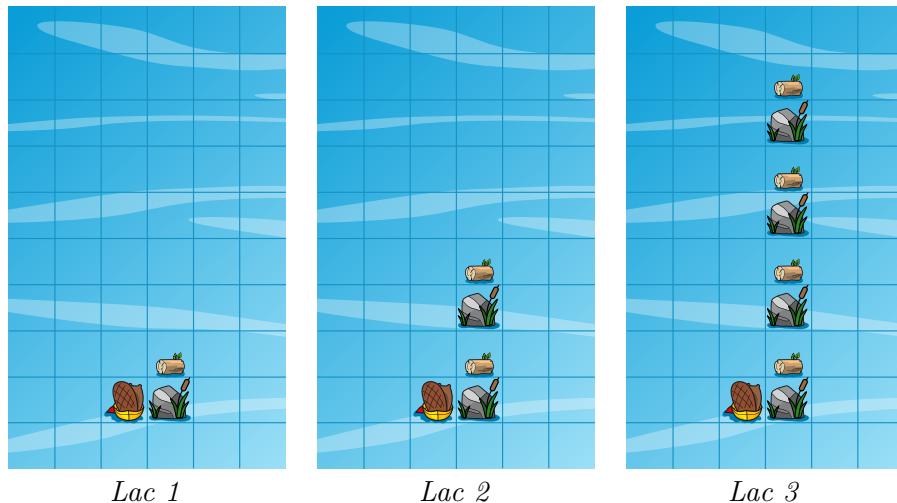
Benno le castor veut ramasser des bûches dans le Seeland. Écris un programme pour que le castor puisse ramasser la bûche dans tous les lacs. Clique sur les cercles sous le lac pour passer d'un lac à l'autre.

Tu peux utiliser ces instructions:

Instruction	Effet
<code>move()</code>	Benno avance d'une case dans la direction où il regarde
<code>turnRight() / turnLeft()</code>	Benno se tourne sur place de 90 degrés sur sa droite ou sur sa gauche
<code>removeLog()</code>	Benno ramasse la bûche de la case où il se trouve
<code>while ...:</code> instruction instruction	Benno répète les instructions décalées tant qu'une condition est remplie.
<code>rockRight()</code>	<i>Condition:</i> Vérifie s'il y a un rocher à la droite de Benno
<code>rockLeft()</code>	<i>Condition:</i> Vérifie s'il y a un rocher à la gauche de Benno

Dans l'exemple suivant, Benno avance d'une case tant qu'il y a un rocher à sa gauche. Quand ce n'est plus le cas, il continue avec l'instruction suivante qui n'est pas décalée — ici, dans l'exemple, `turnLeft()`:

```
while rockLeft():
    move()
    turnLeft()
```



Écris un programme pour ramasser la bûche dans tous les lacs.





A. Auteur·e·s des exercices

James Atlas

Masiar Babazadeh

Leonardo Barichello

Wilfried Baumann

Susanne Berchtold

Maksim Bolonkin

Maria Cepeda

Špela Cerar

Gi Soong Chee

Anton Chukhnov

Darija Dasović

Christian Datzko

Justina Dauksaite

Diane Dowling

Nora A. Escherle

Gerald Futschek

Vernon Gutierrez

Silvan Horvath

Alisher Ikramov

Thomas Ioannou

Asterios Karagiannis

Blaž Kelvišar

David Khachatryan

Doyong Kim

Jihye Kim

Dong Yoon Kim

Vaidotas Kinčius

Jia-Ling Koh

Sophie Koh

Víctor Koleszar

Lukas Lehner

Gunwooong Lim

Yoshiaki Matsuzawa

Hamed Mohebbi

Mattia Monga

Anna Morpurgo

A-Yeong Park

Suchan Park

Elsa Pellet

Jean-Philippe Pellet

Emmanuel Plan

Zsuzsa Pluhár

Wolfgang Pohl

Cesar F. Bolanos Revelo

Pedro Ribeiro

Rokas Rimkus

Kirsten Schlüter

Dirk Schmerenbeck

Jacqueline Staub

Alieke Stijf

Supawan Tasanaprasert

Susanne Thut



Christine Vender

Kyra Willekes

Michael Weigend

Hsu Sint Sint Yee



B. Partenaires académiques

hep/

ABZ

AUSBILDUNGS- UND BERATUNGSZENTRUM
FÜR INFORMATIKUNTERRICHT

Scuola universitaria professionale
della Svizzera italiana

SUPSI

PÄDAGOGISCHE
HOCHSCHULE
ZÜRICH

PH ZH

UNIVERSITÄT TRIER

Haute école pédagogique du canton de Vaud
<http://www.hepl.ch/>

Ausbildungs- und Beratungszentrum für Informatikunterricht
der ETH Zürich
<http://www.abz.inf.ethz.ch/>

La Scuola universitaria professionale della Svizzera italiana
(SUPSI)
<http://www.supsi.ch/>

Pädagogische Hochschule Zürich
<https://www.phzh.ch/>

Universität Trier
<https://www.uni-trier.de/>



C. Sponsoring

HASLERSTIFTUNG

Fondation Hasler

<http://www.haslerstiftung.ch/>



Abraxas Informatik AG

<https://www.abraxas.ch>



Kanton Bern
Canton de Berne

Amt für Kindergarten, Volksschule und Beratung, Bildungs- und Kulturdirektion, canton de Berne

<https://www.bkd.be.ch/de/start/ueber-uns/die-organisation/amt-fuer-kindergarten-volksschule-und-beratung.html>



Kanton Zürich
Volkswirtschaftsdirektion
Amt für Wirtschaft

Amt für Wirtschaft, canton de Zurich

<https://www.zh.ch/de/volkswirtschaftsdirektion/amt-fuer-wirtschaft.html>



Fondation d'Informatique Suisse

<https://informatics-foundation.ch>

cyon

cyon

<https://www.cyon.ch>

senarclens
leu+partner
strategische kommunikation

Senarclens Leu & Partner

<http://senarclens.com/>

UBS

Wealth Management IT and UBS Switzerland IT

<http://www.ubs.com/>

