



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

Quesiti e soluzioni 2016 Tutte le Categorie

<http://www.castoro-informatico.ch/>

A cura di:

Andrea Adamoli, Christian Datzko, Hanspeter Erni

010100110101011001001001
010000010010110101010011
010100110100100101000101
001011010101001101010011
010010010100100100100001

SS! I

www.svia-ssie-ssii.ch
schweizerischervereinfürinformatikind
erausbildung//sociétésuissepourl'infor
matique dansl'enseignement//societàsviz
zera per l'informaticanell'insegnamento



Hanno collaborato al Castoro Informatico 2016

Andrea Adamoli, Christian Datzko, Susanne Datzko, Olivier Ens, Hanspeter Erni, Martin Gugger, Corinne Huck, Carla Monaco, Nicole Müller, Gabriel Parriaux, Jean-Philippe Pellet, Julien Ragot, Beat Trachsler.

Un particolare ringraziamento va a:

Juraj Hromkovič, Giovanni Serafini, Urs Hauser, Tobias Kohn, Ivana Kosírová, Serena Pedrocchi, Björn Steffen: ETHZ

Valentina Dagiene: Bebras.org

Hans-Werner Hein, Wolfgang Pohl, Peter Rossmann: Bundesweite Informatikwettbewerbe (BWINF), Germania

Anna Morpurgo, Violetta Lonati, Mattia Monga: Italia

Gerald Futschek: Austrian Computer Society, Austria

Zsuzsa Pluhár: ELTE Informatikai Kar, Ungheria

Eljakim Schrijvers, Daphne Blokhuis, Marissa Engels: Eljakim Information Technology bv, Paesi Bassi

Roman Hartmann: hartmannGestaltung (Flyer Castoro Informatico Svizzera)

Christoph Frei: Chragokyberneticks (Logo Castoro Informatico Svizzera)

Pamela Aeschlimann, Andreas Hieber, Aram Loosmann, Daniel Vuille, Peter Zurflüh: Lernetz.ch (pagina web)

Andrea Leu, Maggie Winter, Brigitte Maurer: Senarclens Leu + Partner

L'edizione dei quesiti in lingua tedesca è stata utilizzata anche in Germania e in Austria.

La traduzione francese è stata curata da Nicole Müller mentre quella italiana da Andrea Adamoli.



INFORMATIK-BIBER SCHWEIZ
CASTOR INFORMATIQUE SUISSE
CASTORO INFORMATICO SVIZZERA

Il Castoro Informatico 2016 è stato organizzato dalla Società Svizzera per l'Informatica nell'Insegnamento SSII. Il Castoro Informatico è un progetto della SSII con il prezioso sostegno della fondazione Hasler.

HASLERSTIFTUNG

Nota: Tutti i link sono stati verificati l'01.11.2016. Questo quaderno è stato creato il 9 ottobre 2019 col sistema per la preparazione di testi \LaTeX .



I quesiti sono distribuiti con Licenza Creative Commons Attribuzione – Non commerciale – Condividi allo stesso modo 4.0 Internazionale. Gli autori sono elencati a pagina 95.



Premessa

Il concorso del «Castoro Informatico», presente già da diversi anni in molti paesi europei, ha l'obiettivo di destare l'interesse per l'informatica nei bambini e nei ragazzi. In Svizzera il concorso è organizzato in tedesco, francese e italiano dalla Società Svizzera per l'Informatica nell'Insegnamento (SSII), con il sostegno della fondazione Hasler nell'ambito del programma di promozione «FIT in IT».

Il Castoro Informatico è il partner svizzero del Concorso «Bebras International Contest on Informatics and Computer Fluency» (<http://www.bebas.org/>), situato in Lituania.

Il concorso si è tenuto per la prima volta in Svizzera nel 2010. Nel 2012 l'offerta è stata ampliata con la categoria del «Piccolo Castoro» (3^o e 4^o anno scolastico).

Il «Castoro Informatico» incoraggia gli alunni ad approfondire la conoscenza dell'Informatica: esso vuole destare interesse per la materia e contribuire a eliminare le paure che sorgono nei suoi confronti. Il concorso non richiede alcuna conoscenza informatica pregressa, se non la capacità di «navigare» in Internet poiché viene svolto online. Per rispondere alle domande sono necessari sia un pensiero logico e strutturato che la fantasia. I quesiti sono pensati in modo da incoraggiare l'utilizzo dell'informatica anche al di fuori del concorso.

Nel 2016 il Castoro Informatico della Svizzera è stato proposto a cinque differenti categorie d'età, suddivise in base all'anno scolastico:

- 3^o e 4^o anno scolastico («Piccolo Castoro»)
- 5^o e 6^o anno scolastico
- 7^o e 8^o anno scolastico
- 9^o e 10^o anno scolastico
- 11^o al 13^o anno scolastico

Gli alunni iscritti al 3^o e 4^o anno scolastico hanno dovuto risolvere 9 quesiti (3 facili, 3 medi e 3 difficili).

A ogni altra categoria d'età sono stati assegnati 15 quesiti da risolvere, suddivisi in gruppi di cinque in base a tre livelli di difficoltà: facile, medio e difficile. Per ogni risposta corretta sono stati assegnati dei punti, mentre per ogni risposta sbagliata sono stati detratti. In caso di mancata risposta il punteggio è rimasto inalterato. Il numero di punti assegnati o detratti dipende dal grado di difficoltà del quesito:

	Facile	Medio	Difficile
Risposta corretta	6 punti	9 punti	12 punti
Risposta sbagliata	-2 punti	-3 punti	-4 punti

Il sistema internazionale utilizzato per l'assegnazione dei punti limita l'eventualità che il partecipante possa indovinare la risposta corretta.

Ogni partecipante aveva un punteggio iniziale di 45 punti (Piccolo Castoro 27).

Il punteggio massimo totalizzabile era pari a 180 punti (Piccolo castoro 108), mentre quello minimo era di 0 punti.

In molti quesiti le risposte possibili sono state distribuite sullo schermo con una sequenza casuale. Lo stesso quesito è stato proposto in più categorie d'età.



Per ulteriori informazioni:


SVIA-SSIE-SSII Società Svizzera per l'Informatica nell'Insegnamento

Castoro Informatico

Andrea Adamoli

castoro@castoro-informatico.ch

<http://www.castoro-informatico.ch/>

 <https://www.facebook.com/informatikbiberch>



Indice

Hanno collaborato al Castoro Informatico 2016	i
Premessa	ii
1. Cono gelato	1
2. Irrigazione	3
3. Karaoke	5
4. Chi è stato?	7
5. Il codice delle bandiere	9
6. Accoppiamenti	11
7. Cunicoli di formaggio	13
8. Chi vince?	17
9. Scaffale ordinato	19
10. Ricetta segreta	21
11. Lascia che i fiori sboccino!	23
12. Compleanno binario	25
13. Concorrenza (concomitanza)	27
14. Il percorso della biglia	31
15. Sole e fiori	33
16. La chat dei castori	35
17. Quattro commissioni	37
18. Colli di bottiglia	39
19. Messaggi segreti	41
20. Gerarchia	43
21. Costruzione di ponti	45
22. Carte nascoste	47
23. Lavoro di gruppo	49
24. Bomboniere	53



25. Attraverso il tunnel	55
26. Il codice KIX	57
27. Filtro mediano	59
28. Stretta di mano	61
29. Gioco del Nim	63
30. Numeri di maglia ordinati	65
31. Cassy, la tartaruga	67
32. Montacarichi	71
33. Gioco con le biglie	75
34. Due possibilità	79
35. Incrocio di tram	81
36. La codifica delle bandiere	83
37. Lacci da scarpe	85
38. Pittura ricorsiva	87
39. Giocando con le forme	89
40. Scoiattoli egoisti	93
A. Autori dei quesiti	95
B. Sponsoring: concorso 2016	96
C. Ulteriori offerte	98



1. Cono gelato

Nella gelateria LIFO, le palline di gelato richieste vengono impilate (sovrapposte) nel cono esattamente nell'ordine indicato dal cliente.

Cosa deve dire il cliente per ottenere un gelato come quello dell'immagine?

Vorrei un cono con...

- A) ...cioccolato, menta e mirtillo!
- B) ...cioccolato, mirtillo e menta!
- C) ...mirtillo, menta e cioccolato!
- D) ...mirtillo, cioccolato e menta!





Soluzione

La risposta C) è corretta: “Vorrei un cono con mirtillo, menta e cioccolato!”.

Il gusto scelto per primo sarà quello alla base, mentre il gusto scelto per ultimo sarà quello in cima alla pila. Nella risposta A) la pila è ordinata esattamente al contrario. Nelle risposte B) e D) la menta non si trova al centro.

Questa è l'informatica!

L'ordine è importante. Se si elencano i gusti di gelato con un'altra successione, il cono gelato ottenuto è differente.

In informatica si impara presto a capire l'utilità di disporre le cose con ordine e anche che per ogni situazione esiste un ordine adatto.

Senza capire come lavora la gelateria, non è possibile chiedere un determinato cono gelato e, allo stesso modo, senza comprendere appieno una situazione, non è possibile sviluppare un programma appropriato.

La sequenza adottata per questo quesito si chiama “last in, first out” (LIFO) (l'ultimo che entra è il primo ad uscire): l'ultimo gusto scelto è il primo ad essere mangiato.

Siti web e parole chiave

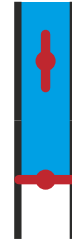
Last In – First Out (LIFO, “ultimo ad entrare, primo ad uscire”), stack (o pila), strutture dati

- <https://it.wikipedia.org/wiki/LIFO>



2. Irrigazione

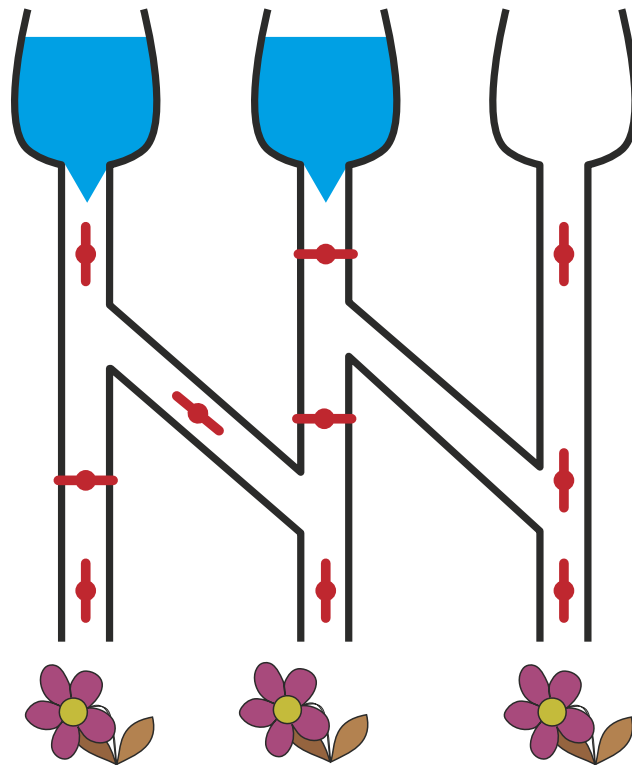
Se la valvola è aperta, l'acqua scorre:



Se la valvola è chiusa, l'acqua non scorre:



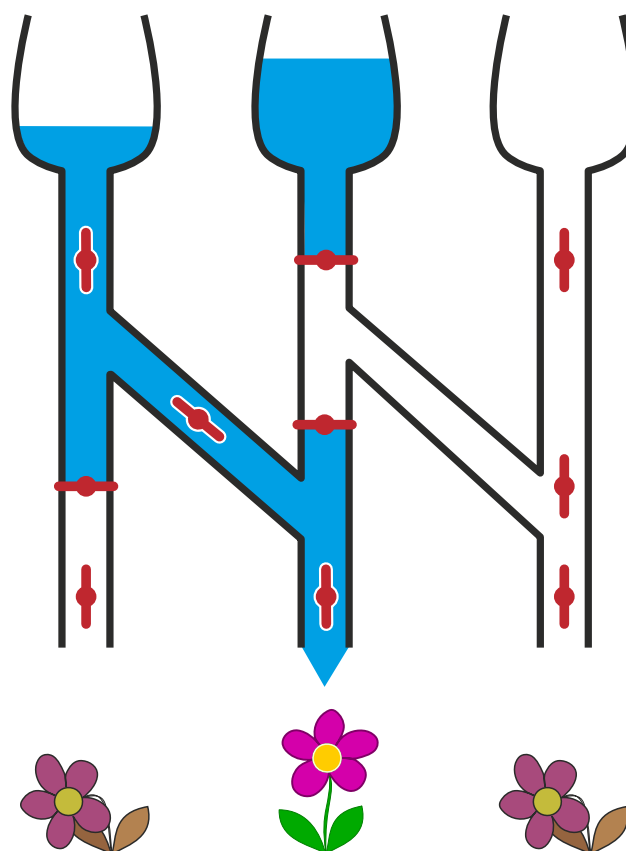
Quale fiore potrà dissetarsi se le valvole sono in questa posizione?





Soluzione

Con le valvole posizionate in questo modo, solo il fiore centrale potrà dissetarsi.



Questa è l'informatica!

Dal punto di vista informatico il sistema di irrigazione è un circuito. Le valvole sono degli interruttori che possono assumere la posizione di “aperto” e “chiuso”. A seconda dello stato dei serbatoi all'entrata (pieno o vuoto) e della posizione degli interruttori, l'informazione “l'acqua scorre” e “l'acqua non scorre” attraversa il circuito, fino ad arrivare ai fiori.

Sugli apparecchi elettronici sono presenti interruttori attraverso i quali scorre l'elettricità. Nei circuiti composti da fibre ottiche le informazioni sono trasmesse con il laser.

Ci sono apparecchi robotici che devono lavorare all'aperto e i loro interruttori elettronici potrebbero danneggiarsi velocemente a causa di forti campi magnetici, dell'elevato tasso di umidità o delle temperature estreme. Questo tipo di robotica contiene dei robusti interruttori nei quali scorre olio per comandi idraulici o aria compressa.

Siti web e parole chiave

Circuiti

- https://it.wikipedia.org/wiki/Circuito_elettronico
- <http://it.wikipedia.org/wiki/Fotonica>
- <http://en.wikipedia.org/wiki/Fluidics>



3. Karaoke

Alcuni castori desiderano cantare il Karaoke insieme. Per fare questo utilizzano quattro programmi diversi. In ogni programma il volume viene regolato in modo diverso.

In quale programma il volume è regolato in modo più alto?





Soluzione

La risposta corretta è C):

- Nel programma “Karaoke Sound 3.0” il volume è a metà.
- Nel programma “AudaBeaverCity” il volume è molto basso.
- Nel programma “Lake Sound Master” il volume è molto alto, vicino al massimo.
- Nel programma “Beaver Karaoke Player 7” l’audio è muto e dunque non si sente nulla.

Questa è l’informatica!

Tu non hai mai visto nessuno dei programmi elencati e ciononostante riesci a capire come impostare il volume. Questo succede perché molti programmi simili condividono gli stessi simboli.

Nel nostro caso, si utilizza un altoparlante stilizzato per rappresentare il controllo del volume. Questo simbolo è presente in tutti i quattro programmi. Esso non è sempre uguale, ma molto simile. Il volume impostato viene di regola rappresentato da una linea o una barra. Se vedi un altoparlante crociato o barrato, capisci subito che l’audio è muto.

I buoni programmi utilizzano sempre simboli tipici per determinate funzioni. È importante, inoltre, che questi simboli siano posizionati dove possono essere trovati facilmente dall’utente. In questo modo ogni utente può utilizzare il programma fin da subito, senza dover leggere dei noiosi manuali d’istruzione.

Siti web e parole chiave

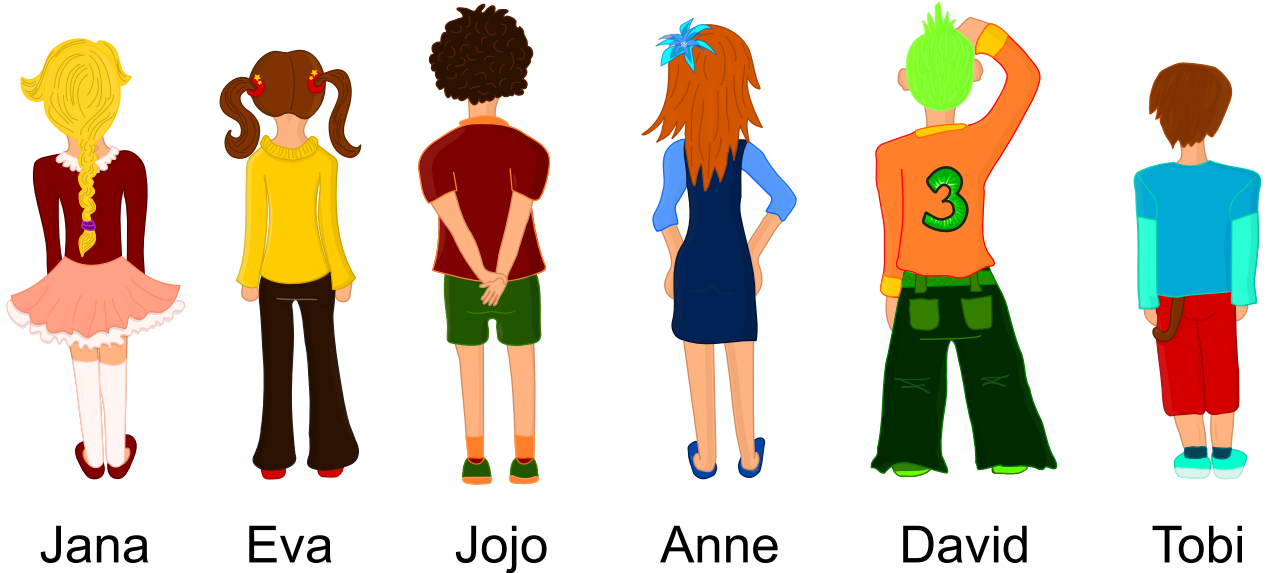
disegno della GUI (“Graphical User Interface” – “Interfaccia Grafica”), interfaccia utente

- https://it.wikipedia.org/wiki/Interface_design



4. Chi è stato?

Sei bambini stanno giocando a pallone all'aperto.



Ad un certo punto: CRAAAACK! Corro verso la finestra e vedo che è rotta. Noto anche il (o la) colpevole che scappa. Aveva capelli lunghi e pantaloni lunghi.

Chi è stato?

- A) Jana
- B) Eva
- C) Jojo
- D) Anne
- E) David
- F) Tobi



Soluzione

La risposta corretta è Eva.

Solo tre bambini hanno capelli lunghi: Jana, Eva e Anne. Due bambini hanno i pantaloni lunghi: Eva e David. Dunque deve essere stata Eva, poiché è l'unica con entrambe le caratteristiche.

Questa è l'informatica!

In informatica si raggruppano spesso oggetti con le stesse caratteristiche (anche dette “proprietà” o “attributi”). Nel nostro esercizio dovevamo considerare due caratteristiche e sei bambini.

Noi cerchiamo la persona con entrambe le caratteristiche. Quando si parla di insiemi di dati questa operazione viene detta “intersezione” di insiemi. Attraverso essa, è possibile “filtrare” gli oggetti con determinate caratteristiche da tutti gli altri (si costruisce dunque un “sotto-insieme”).

Siti web e parole chiave

attributi, criteri di selezione, basi di dati

- [https://it.wikipedia.org/wiki/Oggetto_\(informatica\)](https://it.wikipedia.org/wiki/Oggetto_(informatica))



5. Il codice delle bandiere

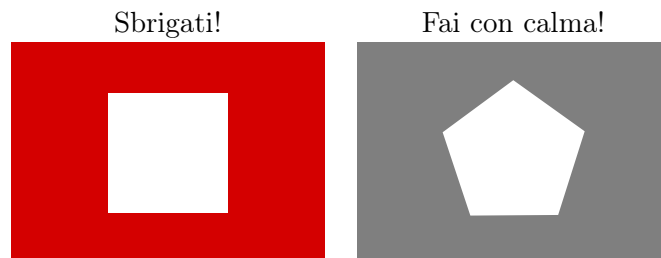
Alberto, il padre di Beatrice e Riccardo, fa il bagnino su una spiaggia. Beatrice e Riccardo giocano con altri bambini sulla stessa spiaggia. Per comunicare con loro, Alberto utilizza un'asta e diverse bandiere. La bandiera posta in alto può avere i seguenti significati:



La bandiera posta in mezzo può avere i seguenti significati:

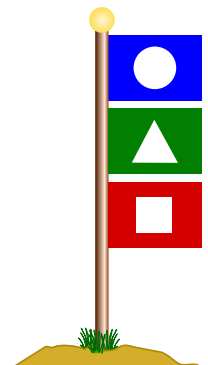


La bandiera posta in basso può avere i seguenti significati:



Cosa significano le bandiere poste su questa asta?

- A) Beatrice, c'è qualcosa da mangiare, sbrigati!
- B) Riccardo, c'è qualcosa da mangiare, sbrigati!
- C) Beatrice e Riccardo, c'è qualcosa da bere, fate con calma!
- D) Beatrice, c'è qualcosa da bere, fai con calma!





Soluzione

La risposta corretta è B). Le tre bandiere, lette dall'alto verso il basso, hanno il seguente significato: la bandiera blu significa che il messaggio è per Riccardo; la bandiera verde che c'è qualcosa da mangiare; la bandiera rossa, che deve sbrigarsi.

Questa è l'informatica!

Utilizzare una sola bandiera per ogni messaggio sarebbe certamente stato meno complicato. Con il metodo di Alberto, però, non solo il colore, ma anche la posizione (in alto, al centro, in basso) delle bandiere è importante. Per questo è necessario dapprima vedere in quale posto le bandiere sono appese e poi, a dipendenza di questo, interpretarne il significato.

Questo principio "Se... allora..." è spesso applicato nell'informatica. Per esempio, negli smartphone (che sono a tutti gli effetti dei computer), il tasto "Home" può avere diversi significati a dipendenza di dove ci si trova. Lo stesso tasto della tastiera ha per un videogame del computer un significato diverso da quello di un editore di testo e, probabilmente, ancora un altro in un programma di elaborazione grafica.

Anche nella programmazione si usa spesso la condizione "Se... allora...". A dipendenza di alcune condizioni indicate ad esempio dall'utente, il computer esegue operazioni diverse.

Siti web e parole chiave

codice, selezioni (istruzioni condizionali), programmazione

- [https://it.wikipedia.org/wiki/Salto_\(informatica\)](https://it.wikipedia.org/wiki/Salto_(informatica))
- [https://it.wikipedia.org/wiki/Codice_\(teoria_dell'informazione\)](https://it.wikipedia.org/wiki/Codice_(teoria_dell'informazione))



6. Accoppiamenti

I castori hanno un nuovo gioco basato su 5 pezzi di puzzle. Alcuni di questi pezzi possono essere accoppiati (a due, a due) quando le loro aree di contatto sono compatibili (ovvero, si possono incastrare).

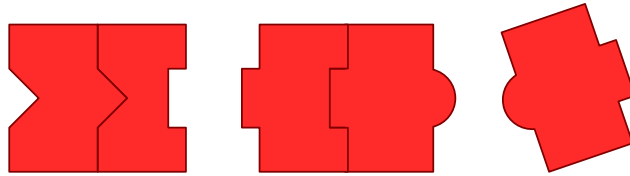
Componi il numero massimo possibile di coppie! Incastra i singoli pezzi di puzzle con il mouse. Attenzione: i pezzi non compatibili non possono essere incastrati.





Soluzione

Si possono creare questi due accoppiamenti:



Se si crea un altro accoppiamento, esso sarà unico e non potranno essere formate altre coppie, lasciando quindi 3 pezzi singoli.

Questa è l'informatica!

Come hai risolto questo compito? Probabilmente hai provato ad accoppiare dei pezzi compatibili e quindi concluso che non c'erano più altre coppie possibili, fino a trovare la soluzione migliore.

Per problemi più complessi, se si utilizza un computer, si può ugualmente provare a trovare tutte le soluzioni possibili. Fin tanto che il compito non è troppo impegnativo, questo funziona. A volte però un approccio simile potrebbe costare molti anni di lavoro anche a un computer. In tal caso, si utilizza spesso una ricerca "euristica". Con essa non vengono provate tutte le soluzioni possibili, ma, attraverso delle semplici regole, si decide in quale direzione muoversi per trovare soluzioni migliori di quella fin lì trovata.

Siti web e parole chiave

metodi forza-bruta (brute-force), euristica, algoritmi greedy

- https://it.wikipedia.org/wiki/Metodo_forza_bruta
- <https://it.wikipedia.org/wiki/Euristica>
- https://it.wikipedia.org/wiki/Algoritmo_greedy



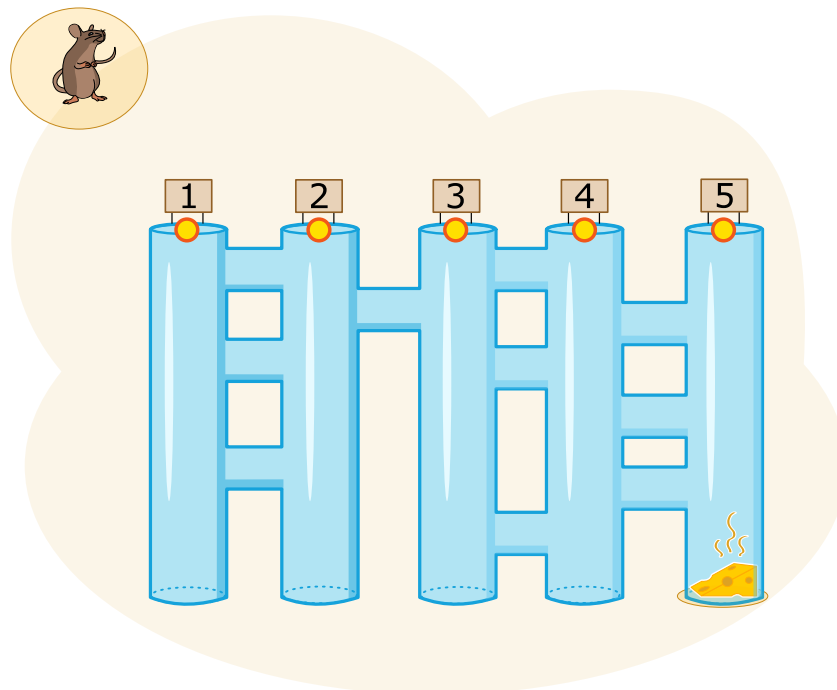
7. Cunicoli di formaggio

Un topo si trova in cima a 5 lunghi cunicoli verticali. Tra i lunghi cunicoli ci sono brevi passaggi orizzontali. Il topo desidera raggiungere il formaggio che si trova in basso, alla fine del lungo cunicolo di destra.

Lungo il cammino, il topo osserva in modo alternato le seguenti istruzioni:

- Cammina verso il basso, lungo il cunicolo verticale fino a quando raggiungi un passaggio orizzontale.
- Cammina attraverso il passaggio orizzontale.

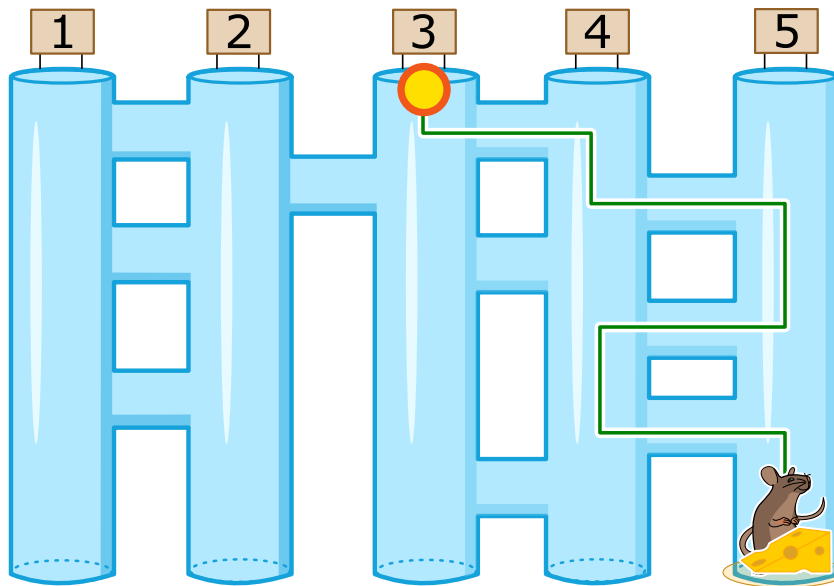
In quale cunicolo verticale deve entrare il topo per raggiungere il formaggio?





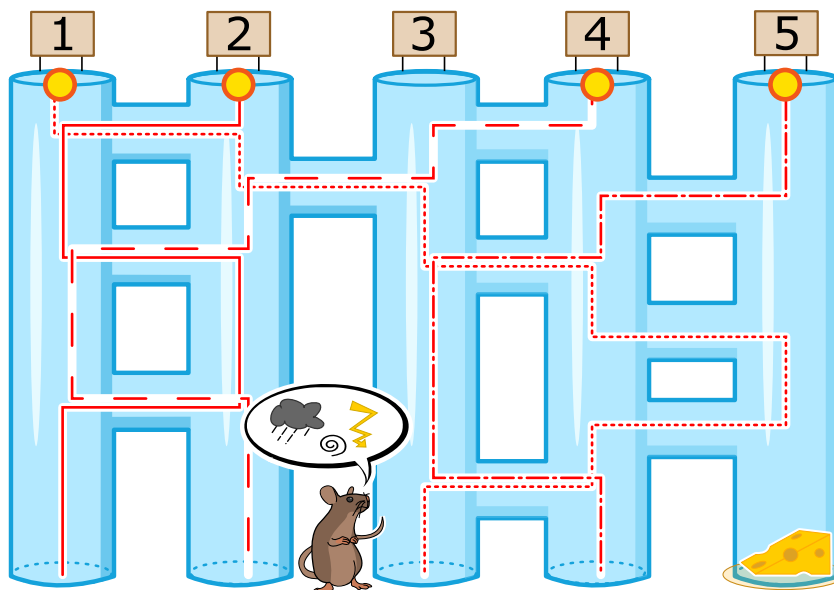
Soluzione

Il topo deve entrare nel cunicolo 3 per raggiungere il formaggio. La linea verde mostra il suo percorso:



Se il topo dovesse imboccare altri cunicoli, non potrebbe mai raggiungere il formaggio. Le linee rosse mostrano i vari percorsi errati:

- Se imboccasse il cunicolo 1, si ritroverebbe alla fine del cunicolo 3.
- Se imboccasse il cunicolo 2, si ritroverebbe alla fine del cunicolo 1.
- Se imboccasse il cunicolo 4, si ritroverebbe alla fine del cunicolo 2.
- Se imboccasse il cunicolo 5, si ritroverebbe alla fine del cunicolo 4.





Questa è l'informatica!

Il topo di questo esercizio può essere paragonato a un computer. Anche i computer ricevono delle istruzioni che seguono scrupolosamente (un insieme di istruzioni viene chiamato “programma”).

Purtroppo un computer non riflette mentre esegue le istruzioni. Il topo del nostro esercizio, una volta imboccato il cunicolo 1, avrebbe potuto facilmente percepire l'odore del formaggio mentre si trovava alla fine del cunicolo 5 ed evitare di prendere il passaggio orizzontale che conduce al cunicolo 4. Ma, dato che anch'egli deve seguire scrupolosamente le istruzioni (proprio come un computer), non lo ha fatto.

Questo significa che quando utilizzi o programmi un computer, esso svolge esattamente le istruzioni che gli impartisci e non quelle che vorresti fossero eseguite.

Siti web e parole chiave

computer, programma

- [https://it.wikipedia.org/wiki/Programma_\(informatica\)](https://it.wikipedia.org/wiki/Programma_(informatica))





8. Chi vince?

Giulia riceve per mail questa pubblicità:



A Giulia piacerebbe molto vincere, però a scuola ha sentito che solo gli adulti possono partecipare a giochi d'azzardo, dove raramente si vince e per giunta i dati personali dei giocatori rischiano di essere usati per scopi illegali.

Cosa deve fare Giulia?

- A) Far finta di essere sua madre e partecipare con i suoi dati personali?
- B) Partecipare utilizzando semplicemente i propri dati personali?
- C) Cancellare la mail?
- D) Chiedere a suo fratello maggiore di partecipare con i suoi dati personali?



Soluzione

La risposta C) è corretta: Giulia dovrebbe cancellare subito la mail visto che si tratta di spam. Le risposte A, B e D richiedono l'indicazione di dati personali per poter partecipare. Questi potrebbero essere indebitamente utilizzati per scopi illeciti, come, per esempio, l'invio mirato di ulteriore spam, o scopi addirittura peggiori.

Questa è l'informatica!

Lo spam costa caro perché ogni giorno milioni di utenti sprecano il loro tempo lavorativo per leggere e cancellare queste mail, creando un danno al sistema economico.

Lo spam è un mezzo di comunicazione di massa spesso utilizzato per distribuire software nocivi (i "malware"). Basta cliccare una sola volta su un'e-mail spam senza disporre di un software di protezione aggiornato, e il proprio computer potrebbe essere controllato a distanza in maniera fraudolenta, ad esempio per diffondere ulteriore spam.

L'informatica non si occupa solo di migliorare i filtri automatici per lo spam e gli altri software di protezione, ma anche di informare la popolazione sui rischi e sulle precauzioni necessarie per navigare in Internet

Tutti i ragazzi, per esempio, dovrebbero sapere che alla base del divieto di partecipare a giochi d'azzardo in Internet vi è un valido motivo, ovvero il fatto che non ci si può certo fidare della casualità del gioco e che si viene sicuramente ingannati. Partecipare quindi è sciocco.

Siti web e parole chiave

spam, dati personali / dati sensibili, software di protezione (antivirus, ...)

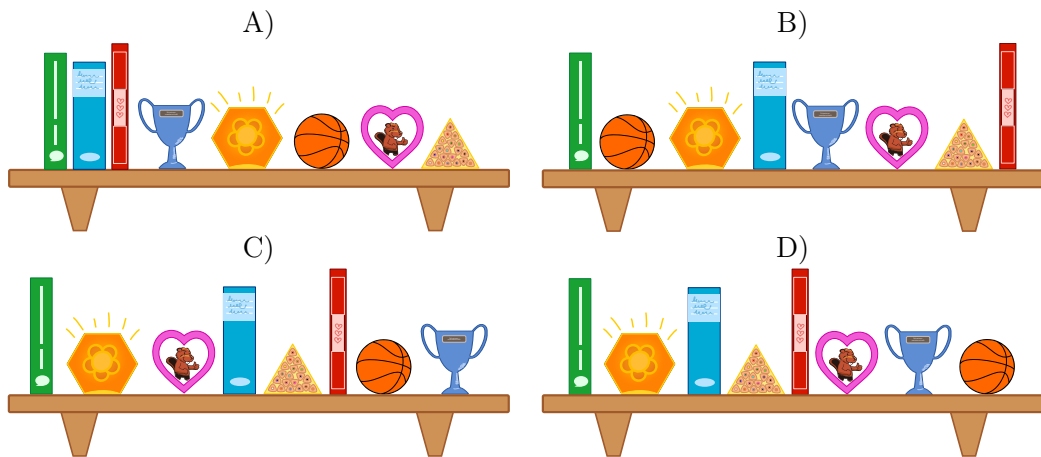


9. Scaffale ordinato

Beatrice riordina il proprio scaffale seguendo 2 regole precise:

1. Oggetti rettangolari non possono stare vicini.
2. Oggetti rotondi non possono stare vicino ad oggetti rettangolari.

Clicca sullo scaffale che osserva le regole di Beatrice!





Soluzione

La risposta D) è corretta:



Lo scaffale A) non è ordinato secondo le regole, poiché a sinistra ci sono 3 libri rettangolari.
Lo scaffale B) non è ordinato secondo le regole, poiché il pallone è posto vicino a un libro.
Lo scaffale C) non è ordinato secondo le regole, poiché il pallone è posto vicino a un libro.
Solo gli oggetti posti sullo scaffale D) seguono le regole.

Questa è l'informatica!

Ammettiamolo: non sempre i nostri scaffali sono in ordine. Questo vale anche per i computer: per esempio, quando su un disco rigido vengono continuamente aggiunti, cancellati o spostati file, si ottengono frammenti sparsi ovunque nel sistema.

Quando questo succede in un disco rigido, per leggere file di grosse dimensioni frammentati si spreca un sacco di tempo. La testina, che si occupa della lettura e della scrittura dai dati, deve saltare da una zona all'altra del disco per recuperare tutti i pezzi. Così, ad esempio, i dati possono essere letti solo con una velocità di 10 megabyte al secondo invece essere letti con velocità di 100 megabyte. In casi del genere si dice che il disco rigido è "frammentato".

Un modo per evitare questa situazione potrebbe essere adottato già al momento della scrittura: i file potrebbero venire salvati in zone del disco vuote, abbastanza grandi per contenere i dati interi. Oppure, ma ciò costa del tempo, si può riorganizzare il disco, ovvero "deframmentarlo".

Il problema della frammentazione non è rilevante per tutti i supporti di memorizzazione. Mentre i dischi rigidi o, in particolare, i nastri hanno difficoltà, SSD e chiavette USB non sono interessati da questo fenomeno, perché non possiedono parti mobili che devono essere spostate per accedere ai dati. Anzi, al contrario: dato che deframmentare significa compiere operazioni di scrittura e che su questi supporti esse sono limitate (ogni chiavetta possiede un numero massimo di possibilità di scrittura e rimozione, dopo sarà inutilizzabile), questo è addirittura controproducente.

Per quanto riguarda le regole seguite da Beatrice ... beh, ognuno è libero di riflettere sulla loro utilità.

Siti web e parole chiave

dati, regole, proprietà

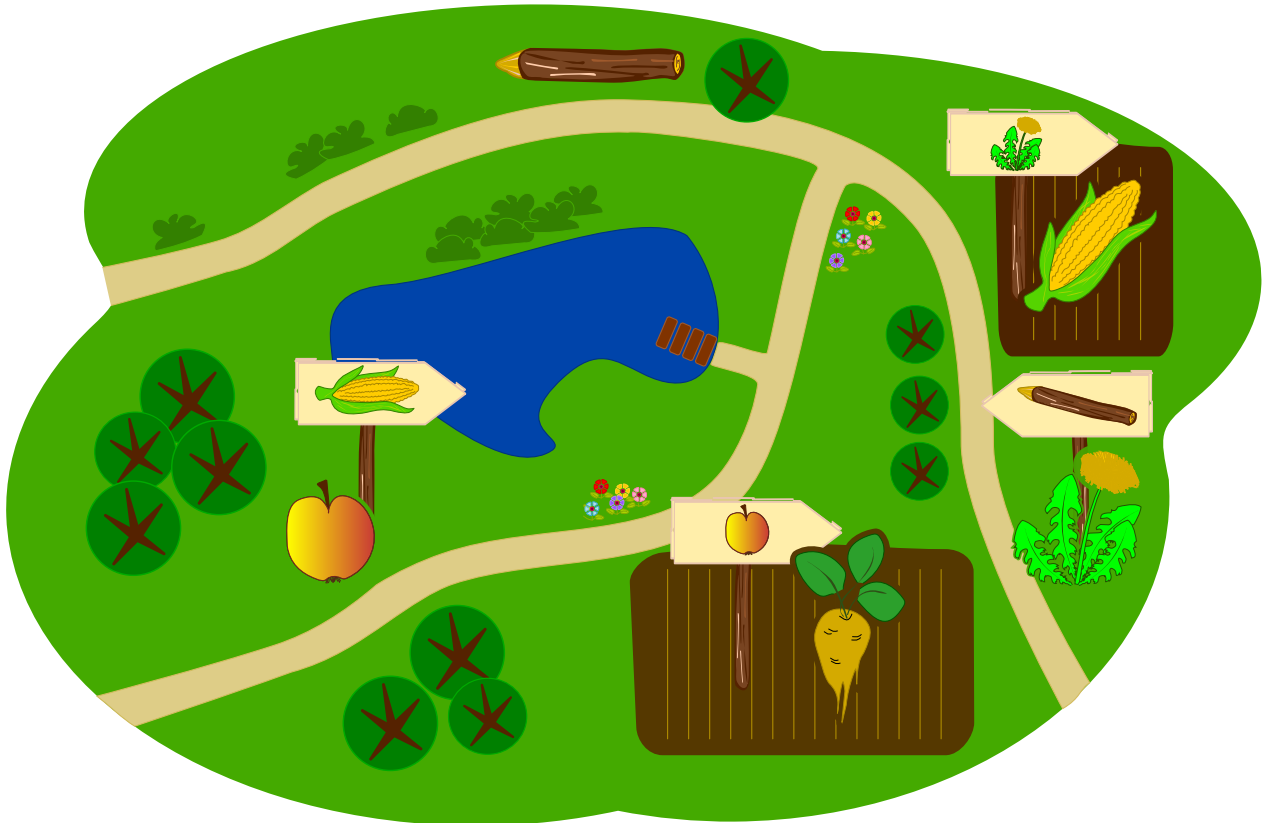
- <https://it.wikipedia.org/wiki/Deframmentazione>



10. Ricetta segreta

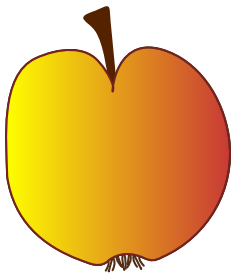
I castori organizzano un party in giardino, per il quale vorrebbero cucinare una torta particolare: il Crunchy-Cake. Purtroppo il loro pasticciere è andato in ferie. I castori, comunque, sanno che tutti i cinque ingredienti devono essere aggiunti nella giusta sequenza.

Fortunatamente in giardino trovano un aiuto lasciato dal pasticciere: vicino ad ogni ingrediente ha piantato un cartello che indica quello successivo.

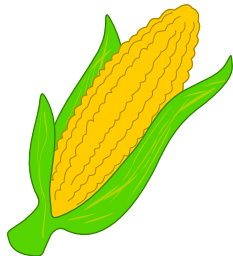


Quale è il primo ingrediente che i castori devono prendere?

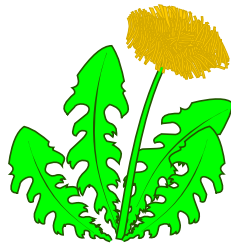
A)



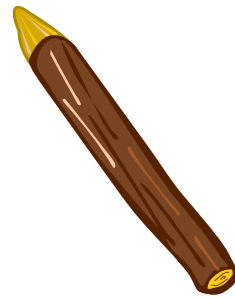
B)



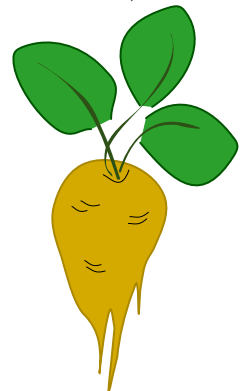
C)



D)






E)





Soluzione

La risposta corretta è E) . Il primo ingrediente può essere solo quello non presente su alcun cartello, poiché questo significa che nessun altro ingrediente viene impiegato prima. Se, ad esempio, la mela  fosse il primo ingrediente, non dovrebbe allora esistere un'indicazione che dalla barbabietola  conduce alla mela.

Questa è l'informatica!

Il pasticciere ha indicato la ricetta per il Crunchy-Cake in modo particolare: per ogni ingrediente ha lasciato un cartello che ne determina il successivo. Questo modo di indicare una sequenza si chiama in informatica *lista concatenata*. Essa è impiegata di frequente, quando da un elemento iniziale si desidera visitare tutti i seguenti in successione. Quando però si conoscono tutti gli elementi, ma non si sa chi è il primo, il tutto diventa più complicato... ..come avrai capito svolgendo il nostro compito. Per questo, quando gli informatici creano delle liste, indicano anche quale è il primo elemento.

Si può anche immaginare che, invece di un solo elemento, ce ne possano essere molti che precedano lo stesso elemento. Questo sarebbe anche il caso della nostra torta, visto che la base e il ripieno possono essere preparati contemporaneamente, non necessariamente in sequenza, e uniti solo alla fine. In questo caso non abbiamo però più una lista bensì un *albero*, poiché diversi percorsi conducono allo stesso elemento finale.

Siti web e parole chiave

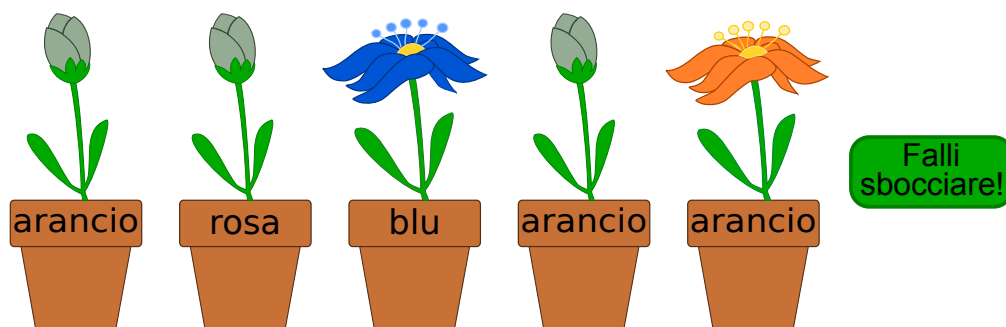
albero, grafo, liste concatenate

- https://it.wikipedia.org/wiki/Lista_concatenata#Liste_semplicemente_concatenate

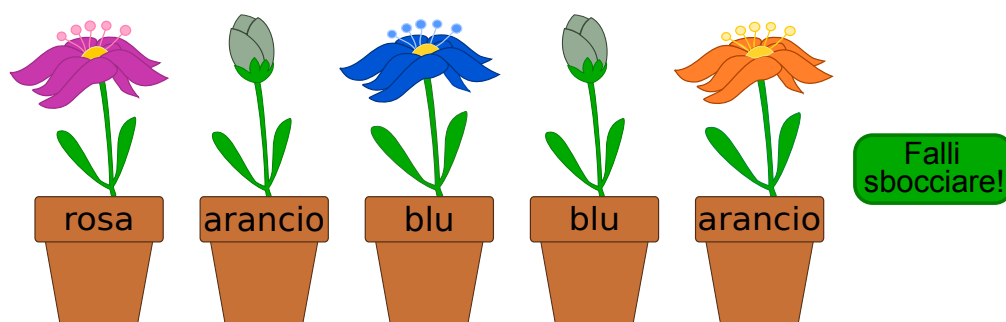


11. Lascia che i fiori sboccino!

Jana gioca con il computer. Segretamente il computer ha scelto i colori per i 5 boccioli. I colori possibili sono blu, arancio o rosa. Tali colori non possono cambiare durante la partita. Jana sceglie dei colori per ogni bocciolo e clicca su “Falli sbocciare”. Solo i boccioli per cui Jana ha scelto il colore giusto sono fioriti, mentre gli altri restano chiusi.



Jana cambia allora i colori per i boccioli non dischiusi. Questa volta ottiene:



Indica il colore di ogni fiore.

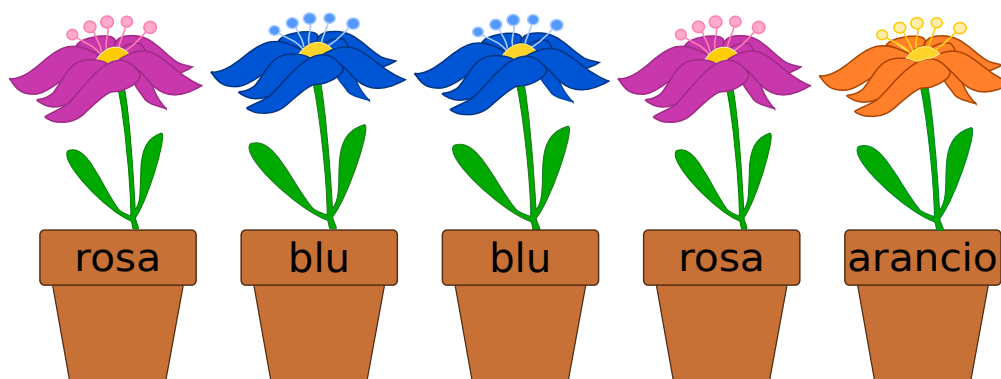


Soluzione

La risposta corretta è “rosa”, “blu”, “blu”, “rosa”, “arancio”. Dopo aver cercato di indovinare due volte, tre boccioli sono fioriti. Dunque possiamo già conoscere il loro colore (primo, terzo e quinto bocciolo da sinistra).

Per il secondo bocciolo, Jana ha dapprima provato con “rosa” e quindi con “arancio”, senza che esso sia fiorito. Siccome esistono solo tre colori possibili, il fiore deve essere necessariamente blu.

Per il quarto bocciolo, Jana ha provato con “arancio” e “blu”. Il fiore può dunque essere solamente “rosa”.



Questa è l'informatica!

Riuscire a trarre delle conclusioni da eventi che sono o non sono accaduti è una capacità importante per risolvere problemi. Il nostro esempio è una versione semplificata di un gioco di strategia molto famoso. Esso è semplificato poiché dopo aver cercato di indovinare, il giocatore riceve tutte le informazioni necessarie sui fiori. Al terzo tentativo tutti i colori possono essere conosciuti con sicurezza ... se si è prestata la dovuta attenzione.

Siti web e parole chiave

logica, giochi di strategia, algoritmo

- <https://it.wikipedia.org/wiki/Mastermind>



12. Compleanno binario

Oggi Ben compie 11 anni. Sua madre, però, ha trovato solo 5 candeline. Per fortuna sa che, anche con 5 candeline, è possibile rappresentare il numero 11. Infila quindi le candeline nella torta una vicino all'altra:

- La candolina più a destra ha valore 1.
- Tutte le altre candeline hanno valore doppio rispetto alla candolina posta alla propria destra.
- I valori delle candeline accese vengono sommati.

Per esempio:



Quali candeline devono essere accese per l'undicesimo compleanno di Ben?





Soluzione

La risposta corretta è A):

A) (01011), il valore delle candele accese è 8, 2 e 1, dunque: $0 \times 16 + 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1 = 11$.



Tutte le altre combinazioni sono errate:

B) (01110), il valore delle candele accese è 8, 4 e 2, dunque: $0 \times 16 + 1 \times 8 + 1 \times 4 + 1 \times 2 + 0 \times 1 = 14$.



C) (10000), il valore della candela accesa è 16, dunque: $1 \times 16 + 0 \times 8 + 0 \times 4 + 0 \times 2 + 0 \times 1 = 16$.



D) (11010), il valore delle candele accese è 16, 8 e 2, dunque: $1 \times 16 + 1 \times 8 + 0 \times 4 + 1 \times 2 + 0 \times 1 = 26$.



E) (11111), tutte le candele sono accese, dunque: $1 \times 16 + 1 \times 8 + 1 \times 4 + 1 \times 2 + 1 \times 1 = 31$.



Questa è l'informatica!

Ogni numero può essere rappresentato in forma binaria. La posizione della candela ne indica il valore. Lo stato della candela (“accesa” o “spenta”), invece, indica se questo valore deve essere sommato oppure no. La stessa cosa succede anche con i numeri binari composti dalle cifre 1 (“candela accesa”) e 0 (“candela spenta”). Il sistema binario è alla base del funzionamento di quasi tutti i computer. Questo per motivi pratici: i circuiti logici che compongono i processori sono molto più semplici da realizzare se basati sul sistema binario che se basati, ad esempio, sul sistema decimale.

Siti web e parole chiave

sistema numerico binario, rappresentazione binaria

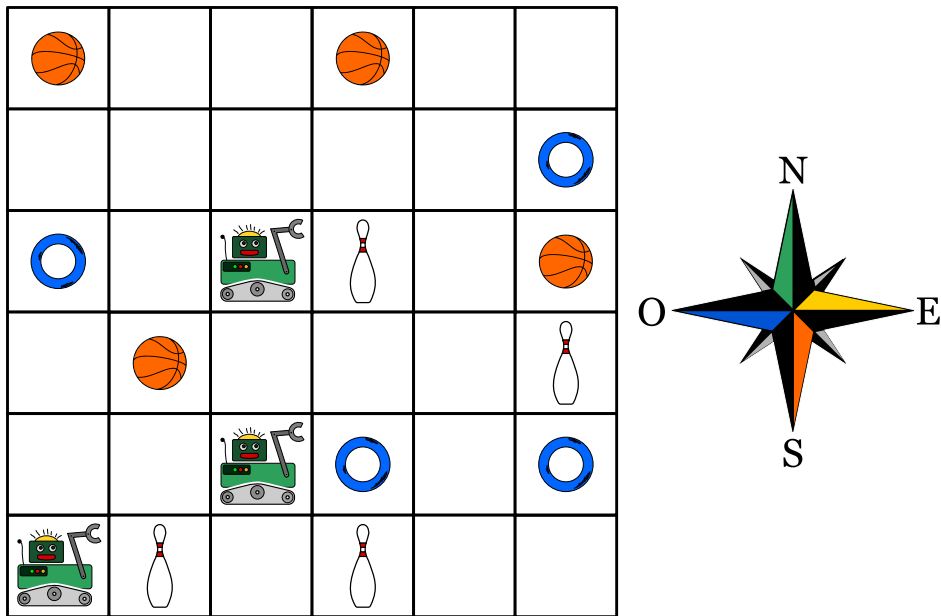
- https://it.wikipedia.org/wiki/Sistema_numerico_binario



13. Concorrenza (concomitanza)

Tre robot lavorano assieme in una squadra. Tu puoi guidare la squadra attraverso gli ordini direzionali N, S, E oppure O. Attraverso questi ordini, indichi la direzione a tutti e tre i robot contemporaneamente: ogni robot si sposterà di una cella nella direzione indicata.

Il tuo compito consiste nel guidare i robot verso l'oggetto che devono prendere. Devi evitare, però, che i robot prendano l'oggetto sbagliato, impartendogli gli opportuni comandi affinché lo aggirino. Esempio: Se guidi i robot con questi ordini "N, N, S, S, O", i robot prenderanno alla fine 2 birilli e un anello.



I robot devono prendere un pallone, un anello e un birillo.

Quali sono gli ordini giusti da impartire?

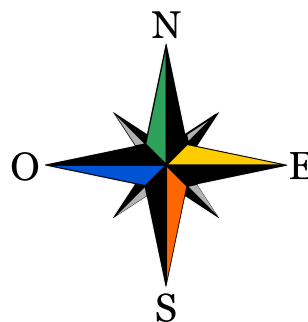
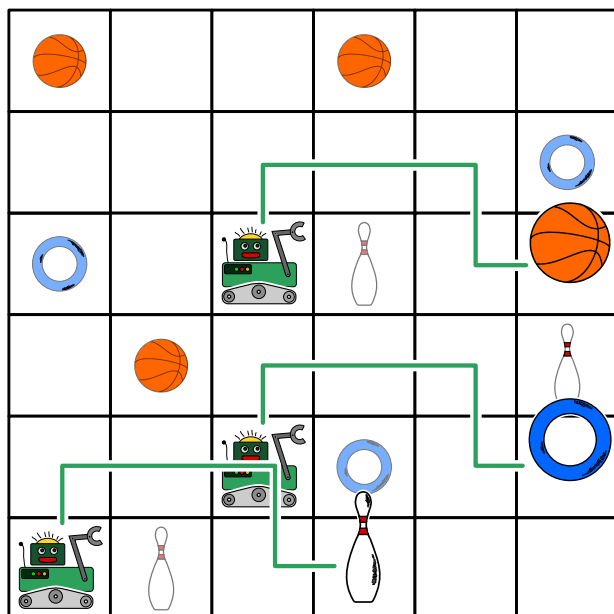
- A) N, E, E, E
- B) N, E, E, S, E
- C) N, N, S, E, N
- D) N, E, E, S, O



Soluzione

La risposta corretta è B):

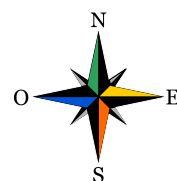
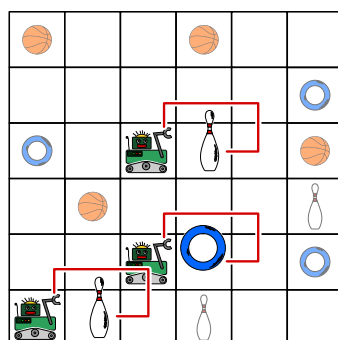
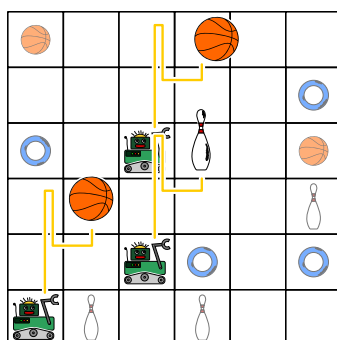
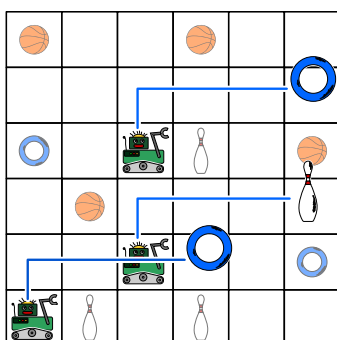
Con gli ordini N, E, E, S, E guidi i robot lungo la linea verde. Alla fine prenderanno un pallone, un anello e un birillo.



La risposta A) è errata: Con gli ordini N, E, E, E guidi i robot lungo la linea blu. Alla fine prenderanno 2 anelli e un birillo.

La risposta C) è errata: Con gli ordini N, N, S, E, N guidi i robot lungo la linea gialla. Alla fine prenderanno 2 palloni e un birillo.

La risposta D) è errata: Con gli ordini N, E, E, S, O guidi i robot lungo la linea rossa. Alla fine prenderanno 2 birilli e un anello.



Questa è l'informatica!

In molti casi i programmi dei computer sono scritti per eseguire dei passi singoli in successione. La maggior parte dei linguaggi e le persone in generale, sono normalmente inclini a elaborare una sola operazione alla volta.

Lo sviluppo dei processori ha raggiunto i propri limiti già da diversi anni, ciò che rende assai difficile un loro incremento delle prestazioni. Allo stesso tempo, però, è diventato più economico l'impiego di molti processori all'interno dello stesso alloggiamento. Per questo motivo, i computer possiedono spesso 2, 4 e anche più processori in grado di lavorare in parallelo. Nelle schede grafiche, ciò è ancora



più estremo: esse possiedono spesso molti “nuclei” computazionali, che sebbene non molto potenti singolarmente, sono in grado di lavorare in parallelo (ovvero assieme) ottenendo ottime prestazioni. In molte schede grafiche è possibile utilizzare i processori anche per operazioni non direttamente connesse al rendering (“rappresentazione”) grafico.

Questo richiede però un cambio nell’approccio alla programmazione: bisogna fare in modo che i processi svolti in parallelo non si intralcino l’un l’altro o che non attendano lungamente i risultati dell’altro per poter svolgere il proprio lavoro. Per ottenere il massimo profitto dalla programmazione parallela, bisogna pianificare le operazioni da svolgere in modo molto accurato.

Siti web e parole chiave

programmazione parallela

- https://it.wikipedia.org/wiki/Calcolo_parallelo
- <https://it.wikipedia.org/wiki/GPGPU>
- [https://it.wikipedia.org/wiki/Concorrenza_\(informatica\)](https://it.wikipedia.org/wiki/Concorrenza_(informatica))

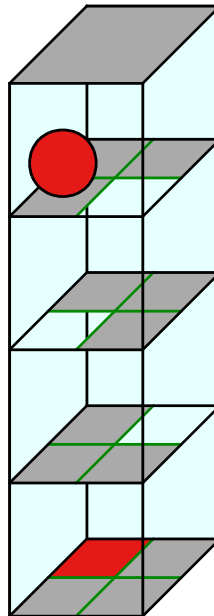
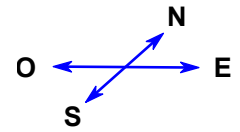




14. Il percorso della biglia

Un labirinto 3D ha 4 piani, ognuno dei quali suddiviso in 4 celle. Una biglia è posizionata nel piano superiore. Nel piano inferiore c'è il traguardo, una cella colorata di rosso.

Tu puoi guidare la biglia con i comandi direzionali N, E, S e O. Attraverso una determinata cella la biglia cade nel piano inferiore. Il labirinto è chiuso, non puoi far passare la biglia attraverso le pareti.



Guida la biglia verso il traguardo!

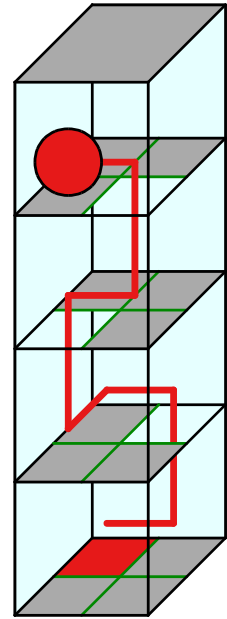


Soluzione

Con la sequenza di comandi “EONEO” (come mostrato con la linea rossa nella figura) oppure “EOENO” la biglia viene guidata al traguardo. Esistono molte altre possibilità per guidare la biglia al traguardo, poiché non è proibito compiere dei percorsi più lunghi e ridondanti.

Questa è l'informatica!

La sequenza di comandi con i quali viene guidata la biglia nel labirinto è di fatto un piccolo programma. Il linguaggio di programmazione associato conosce solo quattro ordini, ovvero N, E, S e O; in informatica essi vengono anche chiamati *istruzioni*. Un programma in questo linguaggio è formato da una *sequenza di istruzioni*, eseguite una dopo l'altra. Anche nei linguaggi di programmazione professionali, la sequenza di istruzioni è un tassello fondamentale di ogni programma. Altri tasselli importanti sono le ripetizioni (cicli) e le istruzioni condizionali (selezioni), così come la frequente chiamata a sotto-programmi detti *procedure*. Con queste semplici strutture è possibile scrivere dei programmi molto complessi.



Siti web e parole chiave

programma, sequenza di istruzioni

- https://it.wikipedia.org/wiki/Programmazione_strutturata



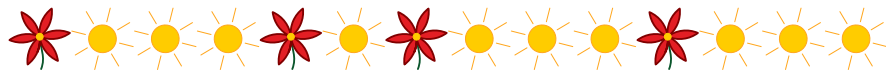
15. Sole e fiori

Barbara ha ricevuto 2 timbri. Uno stampa un fiore, mentre l'altro stampa il sole. Barbara inventa un modo per stampare il proprio nome unicamente con fiori e soli.

Per le diverse lettere utilizza una diversa successione di fiori e soli:

Lettera	B	A	R	E	Y
Successione					

Il suo nome, "Barbara", viene quindi stampato così:



Barbara decide di stampare anche il nome di uno dei propri amici:



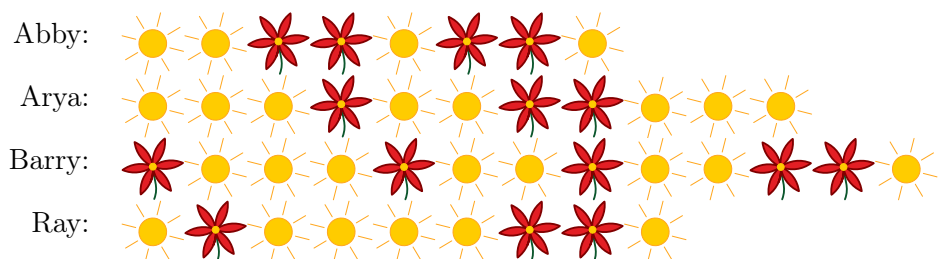
Quale nome ha stampato?

- A) Abby
- B) Arya
- C) Barry
- D) Ray



Soluzione

La risposta corretta è Abby. I nomi degli amici di Barbara hanno i seguenti codici:



Questa è l'informatica!

La codifica dei dati può avvenire in modi diversi. Ad esempio, i simboli digitati attraverso la tastiera sono salvati in codice UTF-8, una variante dell'Unicode. In esso i simboli più comuni (ca. 250 segni) utilizzano 1 byte di memoria, mentre i simboli più rari utilizzano 4 byte: in questo modo è possibile rappresentare milioni di simboli, sufficienti per tutte le lingue della terra.

Il sistema funziona molto bene, però anche tra i simboli più comuni ci sono quelli utilizzati più spesso. Ad esempio la "E" o la "N" sono utilizzati con maggiore frequenza rispetto alla "X" o alla "Z". In questo caso, esistono codici a lunghezza variabile in grado di rendere la memorizzazione ancora più efficiente.

In questi codici a lunghezza variabile è plausibile pensare che il codice di un segno qualsiasi non possa iniziare esattamente con il codice che identifica un altro segno. Solo così la decodifica può avvenire in modo veloce e semplice. Questo tipo di codici vengono detti prefissi. Un esempio molto conosciuto è il codice Morse.

Se si desidera risparmiare più memoria possibile è necessario conoscere la frequenza dei vari simboli nel messaggio da codificare. I segni più comuni avranno codici corti, quelli meno comuni codici più lunghi. Un esempio di questa tecnica è il codice Huffman, un tipo particolare di codice prefisso utilizzato anche nella compressione .zip.

Siti web e parole chiave

codice prefisso, codice Huffman, compressione dei dati

- https://it.wikipedia.org/wiki/Codice_prefisso
- https://it.wikipedia.org/wiki/Codifica_di_Huffman






16. La chat dei castori

La chat dei castori può essere utilizzata gratuitamente e viene finanziata attraverso la pubblicità. L'agenzia di viaggi "Sunshine Travel" ha comprato degli spazi pubblicitari in cui mostra delle immagini diverse per ogni tipo di gruppo bersaglio. Tutti i messaggi scritti in chat vengono analizzati per individuare determinate parole e attribuire loro un punteggio:

- "caro", "cara" o "cari" sono parole introduttive utilizzate dai castori più anziani e ricevono ognuna -2 punti.
- "ciao", "hey" e "bella" sono invece popolari tra i giovani castori e ricevono ognuna +2 punti.
- Le abbreviazioni "bf", "gr8", "np" oder "thx" ricevono +1 punto.
- Ogni parola con 10 o più lettere riceve -1 punti.

Ogni castoro nella chat, in virtù del punteggio ottenuto dai propri messaggi, viene classificato in un determinato gruppo bersaglio:

Punteggio	Gruppo bersaglio	Immagine mostrata
Meno di 0 punti	Anziani	
Più di 0 punti	Giovani	
0	Nessuna classificazione	

Quali immagini vengono mostrate per i seguenti messaggi? Trascina l'immagine corrispondente vicino ad ogni messaggio.

- Cari amici, l'estate è in arrivo e sto cercando una sistemazione comoda vicino al Reno. Grazie per i vostri suggerimenti, Richie.
- Bella! C'è qualcuno?
- @Mia: <3 <3 <3
- Ottimo. gr8. Thx



Soluzione

Messaggio A): Sunshine Travel mostra l'immagine della spiaggia. Il messaggio inizia con "Cari" e contiene le parole "sistemazione" e "suggerimenti" (entrambe con 10 o più lettere), ciò che determina un punteggio negativo.

Messaggio B): Sunshine Travel mostra l'immagine con le tavole da surf. Il messaggio contiene la parola utilizzata dai giovani castori "bella", che ne determina un punteggio positivo.

Messaggio C): Sunshine Travel mostra l'immagine con la torre Eiffel. Il messaggio conta 0 punti in quanto i segni utilizzati non rientrano in alcuna regola.

Messaggio D): Sunshine Travel mostra l'immagine con le tavole da surf. Il messaggio contiene le abbreviazioni "gr8" e "thx" che ne determinano un punteggio positivo.

Questa è l'informatica!

La valutazione dei messaggi secondo determinate regole può essere facilmente eseguita dai programmi per computer. La ricerca di singoli elementi testuali avviene grazie agli algoritmi di pattern-matching ("confronto e ricerca di determinati schemi"), che possono essere applicati sia a testi che a immagini e in altri settori ancora.

Il profiling dell'utente è utilizzato oggi da numerose compagnie su internet, in modo da elaborare automaticamente delle offerte dedicate. In questi casi è importante che gli utenti siano sempre coscienti della problematica e cauti nel fornire i propri dati personali. Gli informatici sono di fatto confrontanti con un dilemma etico: da una parte devono impedire ogni abuso nella raccolta di dati personali, dall'altra questa branca dell'IT fornisce loro interessanti opportunità di impiego e guadagno. Le funzioni per valutare i testi sono importanti anche in altri ambiti, ad esempio per ordinare secondo rilevanza i risultati di una ricerca.

Siti web e parole chiave

user profiling (profilo degli utenti), pubblicità orientata

- https://it.wikipedia.org/wiki/Internet_marketing







17. Quattro commissioni

Durante la pausa di mezzogiorno (dalle 12:00 alle 13:00) Alessandra desidera svolgere le seguenti commissioni:

- comprare un libro in cartoleria
- comprare un litro di latte al negozio di alimentari
- spedire il libro comprato per posta
- bere un caffè al ristorante

Per ogni commissione Alessandra ha calcolato quanto tempo impiega. Questo calcolo è però valido solo per orari al di fuori di quelli di punta, indicati nella tabella qui sotto. Per tanto, Alessandra vuole assolutamente evitare tali orari.

	Luogo	Durata	Orario di punta
	Cartoleria	15 minuti	12:40 – 13:00
	Negozio di alimentari	10 minuti	12:00 – 12:40
	Posta	15 minuti	12:00 – 12:30
	Ristorante	20 minuti	12:30 – 12:50

Trascina le diverse commissioni nella sequenza corretta, in modo che Alessandra possa evitare gli orari di punta.



Soluzione

La sequenza di commissioni corretta è: ristorante, cartoleria, posta, negozio di alimentari. Questo problema possiede alcune limitazioni. Evidenziandole in una tabella, avremmo la seguente rappresentazione (rosso scuro: orario di punta, verde chiaro: orario normale):

Luogo	Durata	12:00-12:05	12:05-12:10	12:10-12:15	12:15-12:20	12:20-12:25	12:25-12:30	12:30-12:35	12:35-12:40	12:40-12:45	12:45-12:50	12:50-12:55	12:55-13:00
Cartoleria	15 minuti					X	X	X					
Negozi di alimentari	10 minuti											X	X
Posta	15 minuti								X	X	X		
Ristorante	20 minuti	X	X	X	X								

Alessandra deve assolutamente passare dalla cartoleria prima delle 12:40 e dal negozio di alimentari dopo le 12:40. Può andare in posta solo dopo aver acquistato il libro in cartoleria e dopo le 12:30, mentre deve andare al ristorante necessariamente prima delle 12:30, poiché alle 12:50 non avrebbe più tempo a sufficienza.

L'unico piano cronologico possibile è dunque (nella tabella indicato con "X"):

- Ristorante 12:00 – 12:20
- Cartoleria 12:20 – 12:35
- Posta 12:35 – 12:50
- Negozi di alimentari 12:50 – 13:00

Questa è l'informatica!

Un compito importante dell'informatica consiste nel trovare soluzioni a determinati problemi, rispettando le limitazioni (condizioni) imposte. Nel nostro caso dobbiamo evitare gli orari di punta dei negozi. In altri problemi con limitazioni ci si pone spesso la domanda se mai esista una soluzione in grado di rispettarle tutte.

Questa domanda in informatica è detta "problema di scheduling" (pianificazione). Uno scheduling è una sequenza corretta o ottimale di operazioni da svolgere per eseguire un determinato compito. In applicazioni industriali, come in grossi progetti o nella produzione, questi problemi sono spesso contemplati. Anche nei computer si utilizza uno scheduler per capire quale risorsa (es., processore) assegnare ad un determinato processo.

Siti web e parole chiave

scheduling, ottimizzazione

- <https://it.wikipedia.org/wiki/Scheduler>

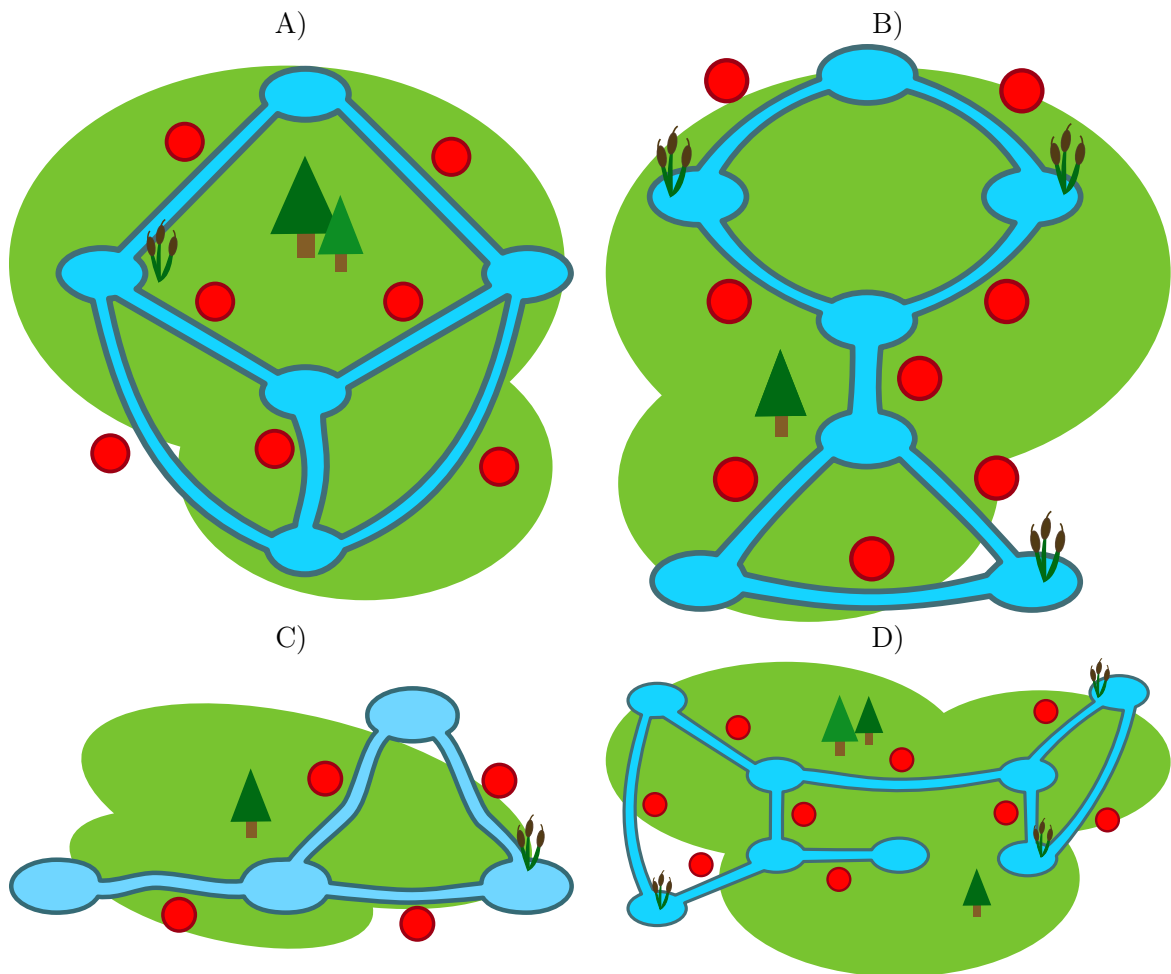


18. Colli di bottiglia

I castori costruiscono delle dighe. Quando nuotano nei propri canali, sono costretti ad aggirare ogni diga uscendo dall'acqua. A loro questo proprio non piace: piuttosto che uscire dall'acqua preferiscono nuotare in altri canali per raggiungere la destinazione.

I canali che devono necessariamente percorrere (perché non esiste un'altra via) per raggiungere la propria destinazione vengono detti "colli di bottiglia". In essi preferiscono non costruire delle dighe. Nelle immagini sono rappresentate alcune zone in cui vivono i castori. I simboli rossi sono i possibili siti per la costruzione di dighe.

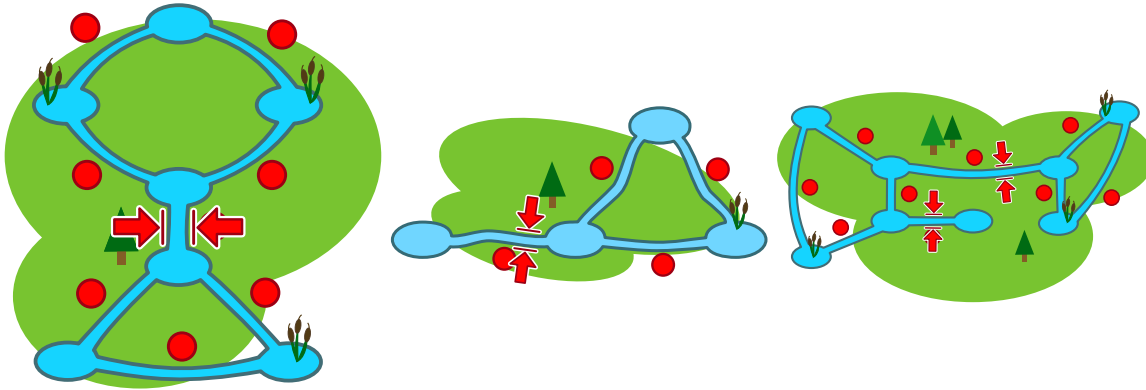
Solo una zona non ha "colli di bottiglia", quale?





Soluzione

La risposta corretta è A). In questa zona esiste una via alternativa per ogni canale. Nella zona B) c'è un collo di bottiglia nel centro. Nella zona C) il collo di bottiglia è a sinistra. Nella zona D) ci sono 2 colli di bottiglia al centro.



Questa è l'informatica!

Le zone in cui vivono i castori sono delle reti di canali e stagni. Esse possono essere facilmente confrontate con Internet, dove computer, telefoni cellulari, televisori, ecc. sono gli stagni, mentre i canali di trasmissione (ponti radio, linee telematiche) sono ... i canali. Originalmente Internet fu pensato per collegare le università negli USA e già allora fu subito importante evitare colli di bottiglia, poiché in caso di un problema sul canale, non esistono alternative.

L'informatica utilizza la teoria dei grafi per studiare i problemi che possono presentarsi sulle reti. I grafi sono definiti da nodi ("stagni") e archi ("canali"). Con i grafi è possibile modellare qualsiasi tipo di rete, come ad esempio quelle di trasporto o quelle di comunicazione. Sono stati sviluppati molti algoritmi per analizzare i problemi di rete. Un problema conosciuto è quello dei "ponti" nei grafi, ovvero trovare quei colli di bottiglia indesiderati all'interno delle reti (come fatto nel nostro esercizio).

Siti web e parole chiave

grafi, ponti, strettoie ("colli di bottiglia")

- [https://it.wikipedia.org/wiki/Ponte_\(teoria_dei_grafi\)](https://it.wikipedia.org/wiki/Ponte_(teoria_dei_grafi))



19. Messaggi segreti

Gli agenti Boris e Berta comunicano attraverso messaggi segreti che nessuno dovrebbe capire all'infuori di loro stessi. Boris desidera inviare a Berta il seguente messaggio:

RITROVOCONBENALLE6

Egli scrive i singoli caratteri del testo nelle celle di una tabella di quattro colonne per cinque righe, procedendo da sinistra a destra e dall'alto verso il basso. Se alla fine restano delle celle vuote, aggiunge degli asterischi (*). L'immagine mostra il risultato di tale operazione:

R	I	T	R
O	V	O	C
O	N	B	E
N	A	L	L
E	6	*	*

Boris genera poi il messaggio segreto nel modo seguente: su un nuovo foglio, riscrive le lettere questa volta seguendo la tabella dall'alto in basso, colonna per colonna da sinistra a destra.

ROONEIVNA6TOBL*RCEL*

Berta utilizza lo stesso metodo per scrivere la risposta segreta a Boris:

SEEAETNNNIAEGCOBVOH*

Cosa ha risposto Berta?

- A) STABENEPOSSOVENIRE
- B) STABENEVENGOANCHEIO
- C) SEVUOIVENGOANCHEIO
- D) VORREIVENIREPUREIO



Soluzione

Il testo iniziale si trova in questo modo: si scrive il messaggio ricevuto nella tabella di quattro colonne per cinque righe, questa volta dall'alto in basso, colonna per colonna a partire da sinistra.

S	T	A	B
E	N	E	V
E	N	G	O
A	N	C	H
E	I	O	*

Si legge poi il messaggio decifrato riga per riga, ottenendo il testo seguente:

STABENEVENGOANCHEIO

L'asterisco alla fine non appartiene al messaggio.

Questa è l'informatica!

I messaggi inviati tramite computer possono essere facilmente intercettati. Quando questi messaggi contengono dati sensibili, come password o informazioni personali, desideriamo che solo il destinatario e nessun altro possa leggere il testo. In questi casi si può cifrare il messaggio in chiaro trasformandolo in un messaggio segreto. Solo il destinatario saprà come decifrarlo per ottenerne il testo originale.

Nella pratica esistono molti metodi per cifrare messaggi. Il metodo applicato in questo esercizio è chiamato trasposizione ed è applicato da ca. 2400 anni. Con esso tutte le lettere del messaggio originale rimangono invariate, ciò che cambia è la loro posizione. Questo metodo è molto semplice da violare, di fatto non si tratta di vera e propria cifratura, ma piuttosto di occultamento dell'informazione.

La crittografia è la scienza che si occupa di studiare metodi per la cifratura e rappresenta un ramo importante dell'informatica. Imprese commerciali e banche presenti su Internet necessitano metodi sicuri di cifratura. Al giorno d'oggi le cifrature sono eseguite dai computer e si basano su metodi matematici, che rendono praticamente impossibile violare i messaggi, se non ne si conosce la chiave.

Siti web e parole chiave

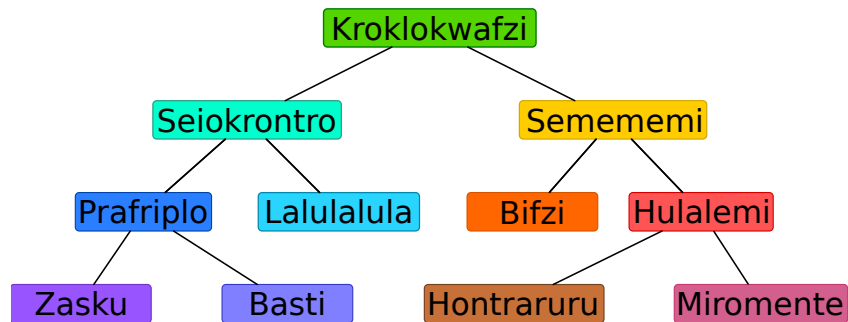
crittografia, cifratura, trasposizione, protezione dei dati

- https://it.wikipedia.org/wiki/Cifrario_a_trasposizione



20. Gerarchia

L'immagine rappresenta le relazioni tra specie animali del pianeta Venere. Una linea che collega due specie indica che tutti gli animali appartenenti alla specie della casella posta più in basso sono anche animali della specie indicata nella casella più in alto.



Per esempio, gli animali della specie “Hulalemi” sono anche della specie “Semememi”. D’altro canto, qualche “Seiokronto” non fa parte dei “Basti”.

Solo una delle seguenti affermazioni è vera, quale?

- A) Tutti i Basti sono anche Seiokronto.
- B) Qualche Hontraruru non è anche Semememi.
- C) Tutti i Zasku sono anche Bifzi.
- D) Tutti i Prafriplo sono anche Basti.



Soluzione

La risposta corretta è A).

- A) Tutti gli animali della specie Basti sono anche della specie Prafriplò. Dato che gli animali della specie Prafriplò sono Seiokrontro, ne consegue che tutti i Basti sono anche della specie Seiokrontro.
- B) Gli Hontraruru sono animali della specie Hulalemi, i quali a loro volta sono dei Semememi. Quindi tutti i Hontraruru sono della specie Semememi.
- C) I Zasku sono anche della specie Prafriplò e non hanno alcuna relazione con la specie dei Bifzi.
- D) Tutti i Basti sono anche della specie Prafriplò, ma non vale il contrario.

Questa è l'informatica!

Le relazioni tra le diverse specie di animali sono rappresentate attraverso uno schema che in informatica è detto "albero". I biologi utilizzano l'albero "filogenetico" per illustrare le relazioni fra le diverse specie.

In informatica si utilizzano spesso degli alberi per rappresentare le relazioni graficamente. Un albero genealogico rappresenta ad esempio le relazioni tra figli, genitori e nonni. Quando si utilizza questa rappresentazione grafica le relazioni tra i vari "nodi" possono essere immediatamente comprese.

Gli alberi sono anche un eccellente mezzo per memorizzare dei dati in modo ordinato e ritrovarli poi velocemente: con pochi passi è possibile accedere in un lampo a enormi insiemi di informazioni.

Siti web e parole chiave

alberi, specializzazione, generalizzazione

- [https://it.wikipedia.org/wiki/Albero_\(grafo\)](https://it.wikipedia.org/wiki/Albero_(grafo))
- https://it.wikipedia.org/wiki/Albero_filogenetico



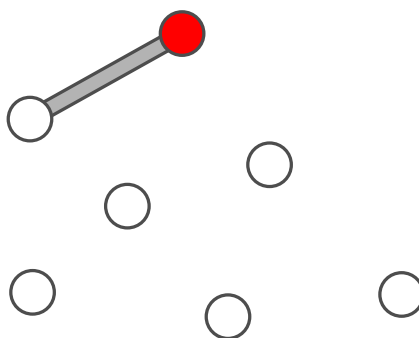
21. Costruzione di ponti

Nonno castoro è diventato timoroso nei confronti dell'acqua. Egli vorrebbe quindi andare dalla propria tana a quella dei familiari attraverso dei ponti. Gli altri castori vogliono molto bene al nonno e per questo desiderano costruire dei ponti che osservino le seguenti regole:

- Il nonno deve poter andare dalla propria tana a qualsiasi altra, attraversando non più di due ponti.
- Oltre al ponte che conduce a una certa tana, possono essercene al massimo altri due che ne escono.

I castori iniziano ad elaborare il progetto. Su un foglio disegnano tutte le loro tane, rappresentandole attraverso dei cerchi. La tana del nonno è un cerchio colorato di rosso. Con una linea pianificano il primo ponte in uscita dalla tana del nonno ... ma poi non sanno più come continuare.

Completa il piano in modo da osservare tutte le regole descritte sopra. Esistono più soluzioni, ma in ogni caso devono essere disegnati altri 5 ponti.

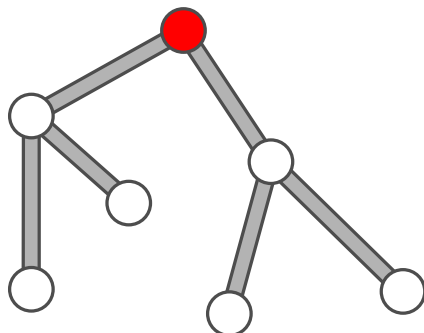




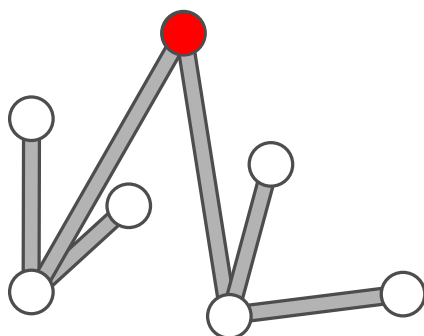
Soluzione

Il progetto esposto qui di seguito soddisfa le regole, poiché:

- dal cerchio rosso si raggiunge ogni altro cerchio attraversando non più di due linee;
- da ogni cerchio escono al massimo 3 linee.



Esistono comunque anche altre soluzioni, ad esempio:



Questa è l'informatica!

Nonostante esistano molte soluzioni, tutte hanno lo stesso schema generale. Dalla tana rossa escono due ponti che si collegano ad altre due tane e, da ognuna di queste, altri due ponti ancora che uniscono le restanti quattro tane. Se ci fosse un'ulteriore tana, il progetto non potrebbe essere realizzato rispettando le regole.

I castori costruiscono quindi un cosiddetto *albero*: tutti i *nod*i (cerchi / tane) possono essere raggiunti percorrendo gli *archi* (linee / ponti). Le regole ci obbligano a utilizzare un tipo particolare di albero: il fatto che solo due archi possono uscire da un nodo fa sì che l'albero sia *binario*; il fatto che dobbiamo percorrere al massimo 2 archi per visitare un nodo qualsiasi fa sì che esso sia *minimale*.

Siti web e parole chiave

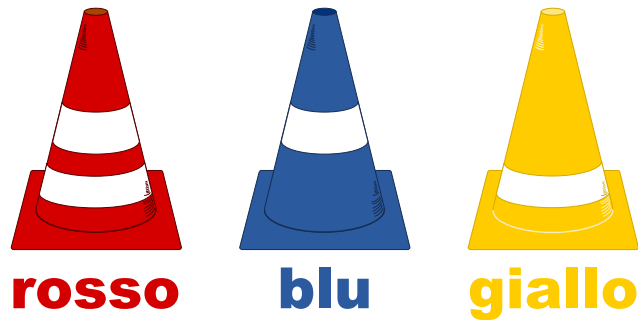
Albero

- [https://it.wikipedia.org/wiki/Albero_\(grafo\)](https://it.wikipedia.org/wiki/Albero_(grafo))
- https://it.wikipedia.org/wiki/Albero_binario



22. Carte nascoste

Vanessa nasconde delle carte sotto a tre coni stradali. Ogni nuova carta viene sempre inserita sotto a un determinato cono. Se il cono nasconde già una carta, quella nuova viene inserita sopra ad essa.



Per descrivere quale carta è stata inserita sotto a quale cono, Vanessa usa la seguente notazione:

rosso ← 5

rosso ← 3



Vanessa ha annotato:

rosso ← 3

giallo ← 5

rosso ← 6

giallo ← 8

blu ← 1

giallo ← 3

Alla fine, quale carta è posta in cima al mazzo nascosto sotto ad ogni cono?

- A) rosso: 3, blu: 1, giallo: 5
- B) rosso: 9, blu: 1, giallo: 16
- C) rosso: 6, blu: 1, giallo: 3
- D) rosso: 8, blu: 1, giallo: 3



Soluzione

All'inizio nessuna carta è stata nascosta sotto ai coni. La tabella seguente mostra quali carte sono inserite in cima, dopo ogni annotazione:

Annotazione	rosso	blu	giallo
	–	–	–
rosso \leftarrow 5	5	–	–
giallo \leftarrow 5	5	–	5
rosso \leftarrow 6	6	–	5
giallo \leftarrow 8	6	–	8
blu \leftarrow 1	6	1	8
giallo \leftarrow 3	6	1	3

È quindi chiaro che C) è la risposta corretta.

Questa è l'informatica!

Il cono stradale può essere interpretato come una variabile, il cui nome è, rispettivamente, “rosso”, “blu” e “giallo”. Una variabile è in pratica uno spazio di memoria (“un cassetto”) del computer in cui poter salvare dei valori (nel nostro caso dei numeri interi). Ogni comando è di fatto un'assegnazione che sostituisce il valore salvato in precedenza con un nuovo valore.

Nei linguaggi di programmazione, le assegnazioni vengono spesso denotate con “=” o “:=”. Questo porta però spesso ad errore di concezione: mentre in matematica, l'espressione “ $x = x + 1$ ” è errata, in informatica significa che la variabile x incrementa il proprio valore di 1 (“il nuovo valore di x è pari al vecchio valore di x più 1”). Spesso, al di fuori dei linguaggi di programmazione, si utilizza la freccia “ \leftarrow ” per indicare un'assegnazione, poiché essa è meglio compresa dai matematici e non scontenta gli informatici.

Siti web e parole chiave

variabile, assegnazione

- [https://it.wikipedia.org/wiki/Variabile_\(informatica\)](https://it.wikipedia.org/wiki/Variabile_(informatica))
- https://it.wikipedia.org/wiki/Operatore_di_assegnamento

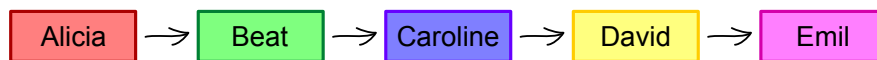


23. Lavoro di gruppo

Sarah e i suoi amici Alicia, Beat, Caroline, David ed Emil devono svolgere un lavoro di gruppo in classe. Essi suddividono il lavoro in modo tale che Sarah abbia il compito di raggruppare i risultati dei compagni. Sarah può ricevere direttamente i risultati del lavoro di Emil, ma per ottenere quelli degli altri deve osservare le seguenti regole:

- Per ricevere i risultati da David, deve prima ottenere quelli di Alicia.
- Per ricevere i risultati da Beat, deve prima ottenere quelli di Emil.
- Per ricevere i risultati da Caroline, deve prima ottenere quelli di Beat e David.
- Per ricevere i risultati da Alicia, deve prima ottenere quelli di Beat ed Emil.

Trascina in nomi nella giusta sequenza, in modo che Sarah possa ricevere tutti i risultati:





Soluzione

La risposta corretta è:

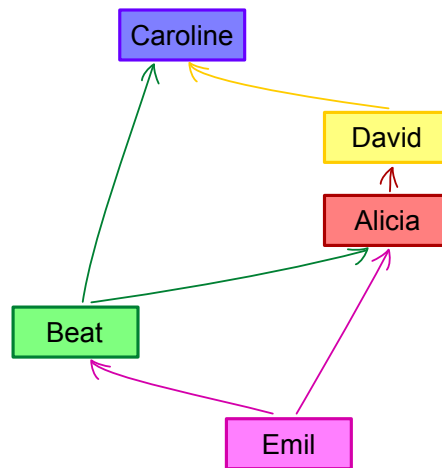


Emil è l'unico a poter consegnare i risultati direttamente, senza aspettare alcun altro compagno. Per questo deve essere posto in cima alla lista. Beat è l'unico compagno che dipende solo da Emil per svolgere il proprio lavoro. Con i risultati di Emil e Beat, Alicia è l'unica a poter svolgere il proprio compito. Analogamente, David è l'unico a poter continuare dopo aver ottenuto i risultati di Alicia e infine anche Caroline può finalmente svolgere il proprio compito.

Questa è l'informatica!

Capita spesso nella vita che alcune cose possano essere portate a termine solo se determinati prerequisiti sono soddisfatti. Anche in informatica ci si occupa di simili operazioni di pianificazione (scheduling). I programmi informatici sviluppati per l'industria hanno spesso a che fare con questi problemi: alcuni processi complicati, come ad esempio la costruzione di automobili, devono essere molto efficienti ed evitare che la catena di montaggio si blocchi per attendere dei pezzi. In questo modo si possono risparmiare molti soldi, sia riducendo il bisogno di immagazzinare parti di auto che incrementando l'utilizzo delle macchine.

Per modellare una pianificazione, si utilizzano dei grafi i cui nodi sono le varie operazioni (nel nostro caso la raccolta dei risultati di un compagno) e gli archi orientati le relazioni di dipendenza. Un grafo del genere viene rappresentato in questo modo:



Ogni possibile soluzione per il problema è rappresentata da percorso attraverso il grafo che visiti tutti i nodi. Nel nostro caso c'è solo un percorso possibile: Emil → Beat → Alicia → David → Caroline.

Trovare una soluzione ad un problema di pianificazione non è però sempre semplice. Esistono delle situazioni complesse in cui, ad esempio, alcune operazioni possono essere svolte in parallelo. In questi casi è anche spesso necessario ottimizzare i costi in termini di tempo o soldi. Di frequente risulta impraticabile individuare e verificare ogni soluzione possibile, per questo vengono utilizzati algoritmi euristici (approssimativi) per individuare soluzioni ottimali.



Siti web e parole chiave

scheduling, pianificazione, grafo orientato

- <https://it.wikipedia.org/wiki/Scheduler>



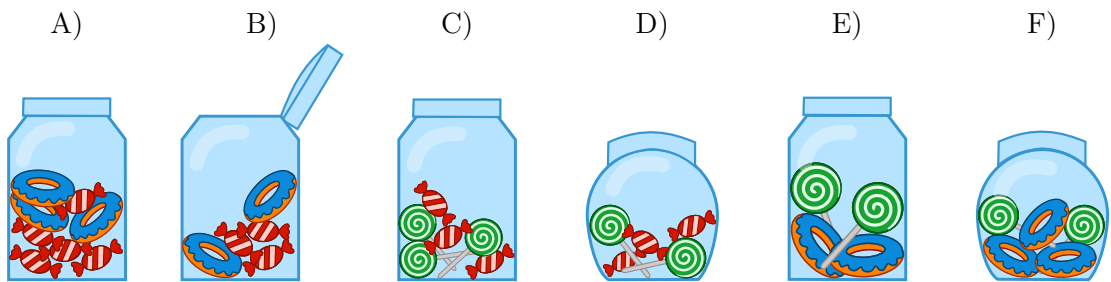


24. Bomboniere

La bomboniera è un contenitore (anche di vetro) per dolci. Carl e Judy hanno 3 bomboniere ciascuno: Carl possiede le bomboniere A), B) e C), mentre Judy possiede le bomboniere D), E) e F). Ogni bomboniera ha le seguenti cinque caratteristiche:

- La bomboniera può essere aperta o chiusa.
- La bomboniera può contenere caramelle a strisce rosse e bianche oppure no.
- La bomboniera può contenere ciambelle blu oppure no.
- La bomboniera può contenere lecca-lecca a spirale verde oppure no.
- La bomboniera può essere tondeggiante oppure angolosa (spigolosa).

Indica la bomboniera che possiede sia le caratteristiche in comune a tutte le bomboniere di Carl che quelle in comune a tutte le bomboniere di Judy.





Soluzione

Le ciambelle blu non devono essere considerate, poiché non sono una caratteristica in comune a tutte le bomboniere di Carl né a quelle di Judy. Esse sono infatti presenti in tutte le bomboniere tranne che in C) e D).

Le due caratteristiche in comune alle bomboniere di Carl A), B) e C) sono:

- ogni bomboniera è angolosa;
- ogni bomboniera contiene caramelle a strisce rosse e bianche.

Le due caratteristiche in comune alle bomboniere di Judy D), E) e F) sono:

- ogni bomboniera è chiusa;
- ogni bomboniera contiene lecca-lecca a spirale verde.

Ora, solo la bomboniera C) possiede tutte le quattro caratteristiche: è angolosa, contiene caramelle a strisce rosse e bianche, contiene lecca-lecca a spirale verde ed è chiusa.

Questa è l'informatica!

Nei propri modelli di dati, l'informatica predilige raggruppare gli oggetti con caratteristiche (meglio, "proprietà" o "attributi") comuni. Nel nostro quesito abbiamo a che fare con 5 proprietà e 2 due gruppi. Il compito consiste nel cercare oggetti che posseggano tutte le proprietà che i due gruppi hanno rispettivamente in comune. Nelle basi di dati relazionali (e in matematica), questa operazione viene detta "intersezione di insiemi".

Nelle basi di dati relazionali, si possono identificare (o, meglio, *filtrare*) velocemente degli oggetti con determinate proprietà in insiemi enormi ("costruzione di sotto-insiemi"). Per esempio, nei negozi on-line è possibile filtrare gli smartphone con determinate proprietà per accumulatori, display o altri elementi ancora.

Nella costruzione delle basi di dati è importante predisporre in modo dettagliato quali proprietà utilizzare nel modello dei dati. Se importanti proprietà dovessero essere dimenticate, sarà difficile in futuro selezionare l'oggetto con le caratteristiche volute. Se, invece, dovessero essere contemplate delle proprietà di scarsa rilevanza, si renderebbe la base di dati inutilmente complessa e costosa.

Siti web e parole chiave

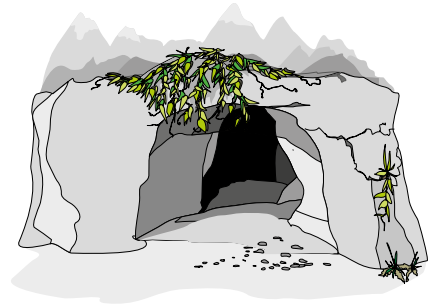
orientamento agli oggetti, attributi, logica

- [https://it.wikipedia.org/wiki/Intersezione_\(insiemistica\)](https://it.wikipedia.org/wiki/Intersezione_(insiemistica))
- [https://it.wikipedia.org/wiki/Unione_\(insiemistica\)](https://it.wikipedia.org/wiki/Unione_(insiemistica))
- https://it.wikipedia.org/wiki/Teoria_degli_insiemi
- https://it.wikipedia.org/wiki/Base_di_dati




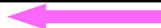

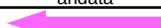
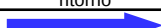
25. Attraverso il tunnel

Anna e Ben fanno una passeggiata con i loro genitori. Lungo il cammino c'è un tunnel. Per esperienza sanno che ognuno di essi impiega un tempo diverso a percorrerlo: Anna impiega 10 minuti, Ben 5 minuti, la madre 20 minuti e il padre 25 minuti. Attraverso lo stretto e scuro tunnel possono passare solo una o, al massimo, 2 persone. Essi devono compiere quindi molti passaggi. Quando si attraversa il tunnel in due, si impiega tanto tempo quanto la persona più lenta. Nel tunnel bisogna assolutamente utilizzare una torcia.



Quando la famiglia si trova all'entrata del tunnel, si accorge che l'accumulatore della torcia ha energia solo per altri 60 minuti. Rusciranno a passare tutti attraverso il tunnel prima che la torcia si spenga? Anna ne è certa: "Sicuro, possiamo farcela in 5 passaggi!"

Sposta i nomi nelle celle appropriate, in modo da attuare il piano di Anna.

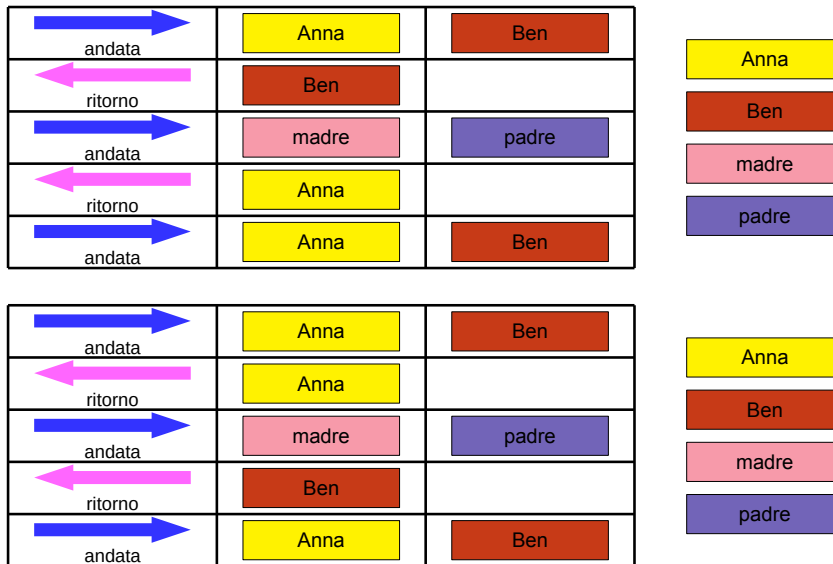
 andata		
 ritorno		
 andata		
 ritorno		
 andata		

Anna
Ben
madre
padre



Soluzione

Passare tutti attraverso il tunnel in non più di 60 minuti è possibile solo se il padre e la madre (i più lenti) compiono il tragitto una sola volta. Questo è realizzabile se dapprima Anna e Ben lo attraversano e poi uno dei due torna indietro per portare la torcia ai genitori. Essi attraverseranno il tunnel assieme e il fratello rimanente porterà di nuovo indietro la torcia per scortare il primo all'uscita. Con questo metodo è indifferente se il primo a tornare indietro è Ben oppure Anna: entrambi dovranno tornare una volta, impiegando complessivamente lo stesso tempo.



Questa è l'informatica!

In informatica succede spesso che ci siano risorse limitate per risolvere un problema. Allo stesso tempo la soluzione trovata deve anche soddisfare delle condizioni accessorie. Il problema viene risolto delineando un piano che tiene conto dell'utilizzo delle risorse e si cura di soddisfare le condizioni accessorie.

In informatica problemi simili vengono risolti con algoritmi di *scheduling*. Dato che i problemi sono molto complessi, il computer impiega anche molti anni per calcolare una soluzione ottimale. Questa soluzione non deve essere necessariamente la migliore, basta che soddisfi tutte le condizioni. Anche nel nostro caso, il problema richiedeva il passaggio del tunnel entro 60 minuti e non il passaggio nel più breve tempo possibile. Il fatto che la soluzione proposta fosse comunque la migliore è solo un caso: l'importante, come detto, era trovarne una ottimale ... e poi passare al prossimo quesito.

Siti web e parole chiave

(job) scheduling, condizioni, risorse

- <https://it.wikipedia.org/wiki/Scheduler>



26. Il codice KIX

In Olanda i numeri di avviamento postale sono composti da 4 caratteri che possono essere lettere o cifre. Esiste in aggiunta anche un codice a barre associato, detto codice KIX. Ogni simbolo del codice KIX è suddiviso in una parte superiore (due barre lunghe e due corte) e una parte inferiore (di nuovo due barre lunghe e due corte). Per ogni carattere si uniscono queste due parti al centro. Nella tabella viene mostrato il codice KIX composto per 0, 7, G e Y:

	0	1	2	3	4	5
	6	7	8	9	A	B
	C	D	E	F	G	H
	I	J	K	L	M	N
	O	P	Q	R	S	T
	U	V	W	X	Y	Z

Il codice KIX per il numero di avviamento postale G7Y0 è quindi:

A quale numero di avviamento postale corrisponde questo codice KIX:



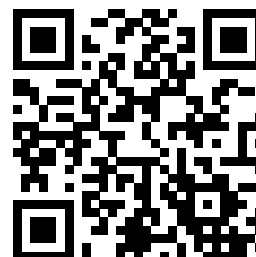
Soluzione

La risposta corretta è BC16:

	0	1	2	3	4	5
	6	7	8	9	A	B
	C	D	E	F	G	H
	I	J	K	L	M	N
	O	P	Q	R	S	T
	U	V	W	X	Y	Z

Questa è l'informatica!

In Olanda si utilizza effettivamente il codice KIX. Codici facilmente leggibili dalle macchine permettono, ad esempio, di inviare pacchetti o lettere automaticamente. Codici simili sono spesso utilizzati: codici a barre (sui prodotti) per la scansione alla cassa, o codici QR, sviluppati dall'industria dell'auto per riconoscere le singole parti e oggi utilizzati come veicolo pubblicitario (esistono infatti molte applicazioni per smartphone che permettono di decifrarli). Cosa pubblicizza il codice QR posto qui a lato?



Siti web e parole chiave

codice KIX, codice a barre, codice QR

- <https://nl.wikipedia.org/wiki/KIX-code>
- https://it.wikipedia.org/wiki/Codice_a_barre
- <https://en.wikipedia.org/wiki/RM4SCC> (inglese)

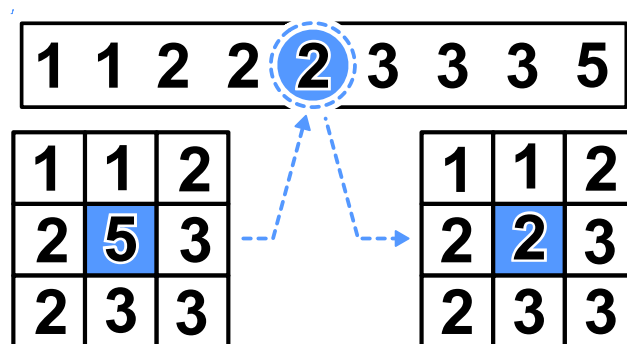


27. Filtro mediano

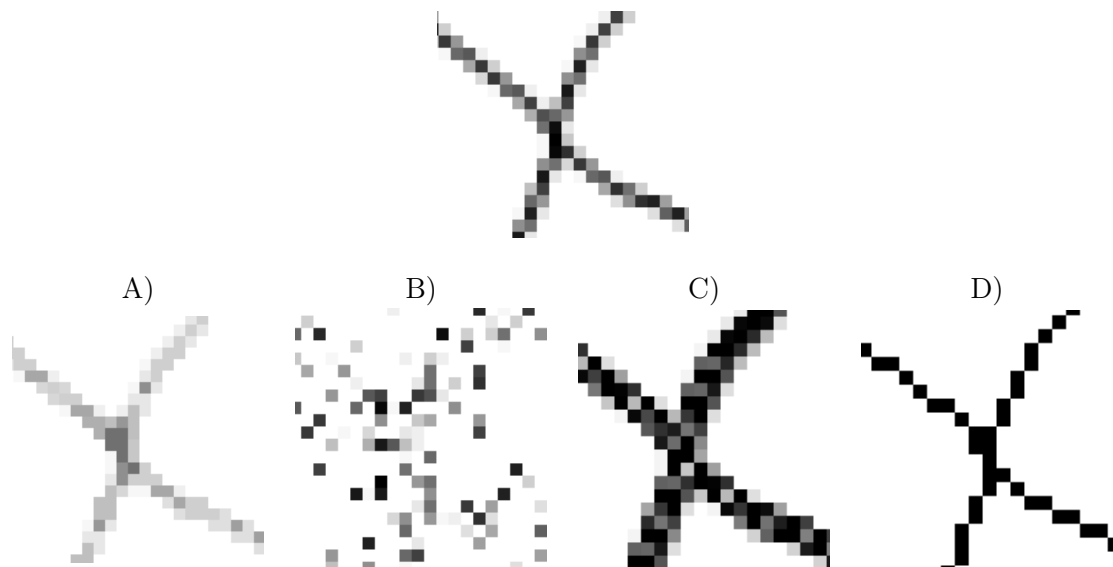
Un'immagine viene salvata come tabella indicante il tono di luminosità (da 1 a 5) per ogni pixel. Il valore 1 indica il nero, mentre il valore 5 indica il bianco. I valori tra 2 e 4 indicano toni di grigio sempre più chiari. Un cosiddetto "filtro mediano" modifica i toni di grigio di ogni pixel dell'immagine, in modo che...

- ... il valore del pixel e quello degli otto vicini vengono riportati in una sequenza ordinata...
- ... il pixel assume il valore del quinto elemento della sequenza (la mediana) quale nuovo tono di grigio.

In questo esempio, il pixel al centro passa dal valore 5 al valore 2:



Che apparirà questa immagine dopo aver applicato il filtro mediano?





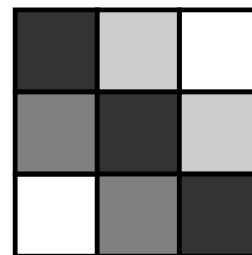
Soluzione

La risposta corretta è A):

Attraverso il filtro mediano vengono elaborati parti di immagine composte da nove pixel. I pixel neri al centro ottengono nuovi valori. Osservando l'esempio a destra è chiaro che solo 3 pixel possiedono una tonalità nera, la mediana è dunque necessariamente più chiara rispetto al pixel originale. Il nero sarà dunque convertito in un tono di grigio più chiaro.

Nell'immagine, in qualsiasi spezzone di 9 pixel, i "neri" sono sempre in minoranza, quindi l'immagine filtrata non potrà possedere toni di nero.

Questo si verifica solo nell'immagine A).



Questa è l'informatica!

Con l'elaborazione di immagini si vogliono applicare velocemente degli effetti visivi. Spesso si desidera ottenere immagini più vive o a fuoco. Altre volte si vogliono inserire effetti artistici per personalizzare le proprie foto. Tutti questi effetti si ottengono grazie a filtri dell'immagine.

Un esempio di tali filtri è quello mediano. Esso viene ad esempio impiegato per correggere errori nelle foto, dovuti magari a un piccolo difetto nei sensori di acquisizione. In seguito a questo effetto, l'immagine risulta più smussata e ogni pixel viene uniformato. Grazie a questo filtro possono essere dunque escluse determinate forme di "rumore".

Siti web e parole chiave

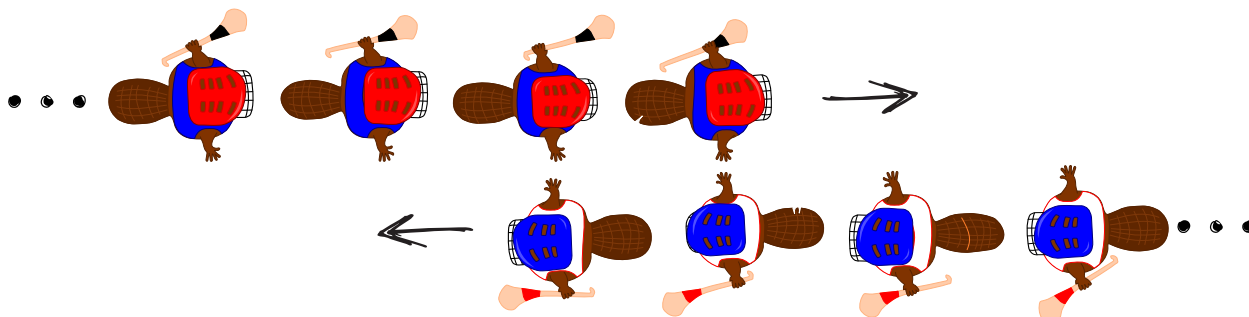
elaborazione dell'immagine, filtro mediano, livelli di grigio

- https://it.wikipedia.org/wiki/Filtro_mediano
- https://it.wikipedia.org/wiki/Livelli_di_grigio
- [https://it.wikipedia.org/wiki/Rumore_\(immagine\)](https://it.wikipedia.org/wiki/Rumore_(immagine))



28. Stretta di mano

I castori giocano volentieri all'Hurling, lo sport nazionale dell'Irlanda. Alla fine della partita i giocatori di ogni squadra si allineano in fila, uno dietro all'altro. Essi si dirigono poi verso la fila della squadra avversaria e si stringono la mano dicendo "Grazie per la partita!".



La stretta di mano funziona in questo modo: dapprima si stringono la mano i giocatori posti in cima alla propria fila, poi essi stringono la mano al giocatore che occupa il secondo posto della fila avversaria (come mostrato nella figura). Si continua in questo modo fino a quando anche gli ultimi giocatori della fila hanno stretto la mano a tutti quelli della fila avversaria.

Nell'Hurling ci sono 15 giocatori per ogni squadra. Per la stretta di mano tra due giocatori e lo spostamento verso il prossimo giocatore avversario si impiega 1 secondo.

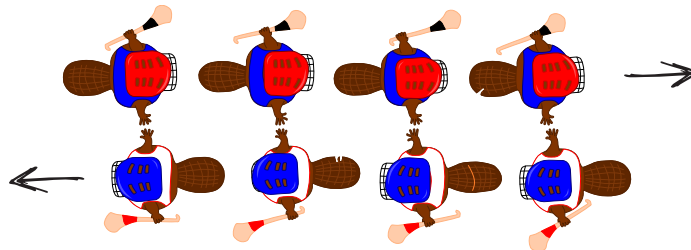
Quanti secondi si impiegano in totale per la stretta di mano tra tutti i membri delle due squadre?



Soluzione

La risposta corretta è 29.

La stretta di mano funziona così: il primo giocatore di ogni fila stringe la mano a tutti gli altri giocatori. Con 4 giocatori si impiegherebbero 4 secondi e si avrebbe la seguente situazione:



A quel punto, l'ultimo giocatore di ogni squadra avrebbe stretto la mano ad un giocatore avversario per la prima volta. Egli dovrebbe quindi stringere la mano ancora a tutti i restanti giocatori della fila avversaria, impiegando altri 3 secondi.

Generalizzando, possiamo quindi affermare che con n giocatori, impieghiamo dapprima n secondi e poi ancora $n - 1$ secondi. In totale abbiamo dunque $n + n - 1 = 2n - 1$ secondi. Per 15 giocatori si impiegano allora $15 + 15 - 1 = 29$ secondi per stringersi la mano.

Questa è l'informatica!

Per le squadre dell'Hurling composte da 15 giocatori, possiamo calcolare esattamente quanto tempo impiega complessivamente la stretta di mano. Quanto impiegherebbero, invece, due squadre di Hockey con 22 giocatori ciascuno? Possiamo ancora utilizzare l'algoritmo impiegato per l'Hurling? Sarebbe davvero utile poter disporre di un algoritmo in grado di valutare il tempo di esecuzione in generale, senza dover esaminare in dettaglio ogni caso.

L'informatica ricerca intensamente algoritmi generali per calcolare il tempo di esecuzione. Le analisi effettuate forniscono delle espressioni matematiche contenenti una variabile n associata alla grandezza dei dati. Per il nostro esempio, otteniamo questa espressione sostituendo il "numero di giocatori" (15) con n : $2n - 1$. In questo modo possiamo facilmente calcolare il tempo di esecuzione anche per squadre con più o meno giocatori. Per 22 giocatori, impiegheremmo $2n - 1 = 2 \cdot 22 - 1 = 43$ secondi, per 40 giocatori 79 secondi e così via.

L'espressione $2n - 1$ è una funzione lineare. L'algoritmo della "stretta di mano" appartiene quindi alla classe degli algoritmi con tempi di esecuzione lineari, designati con $O(n)$. Come sarebbe, però, se la stretta di mano avvenisse in modo diverso? Per esempio, se i giocatori si stringessero la mano singolarmente (e non in contemporanea). Ebbene, questo algoritmo appartenerrebbe alla classe $O(n^2)$ e le squadre di Hurling impiegherebbero $15^2 = 225$ secondi (ca. 4 minuti). Se l'algoritmo avesse poi un tempo di esecuzione esponenziale, ovvero se fosse della classe $O(2^n)$ si impiegherebbero $2^{15} = 32768$ secondi, ovvero ca. 9 ore. In questo caso gli spettatori sarebbero tornati a casa da molto tempo. E quindi utile, in questi casi, domandarsi se non sia possibile svolgere determinate operazioni in parallelo per risparmiare tempo.

Siti web e parole chiave

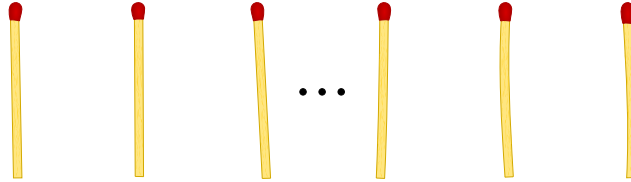
complessità temporale, analisi del tempo di esecuzione (run-time)

- <https://it.wikipedia.org/wiki/Run-time>
- https://it.wikipedia.org/wiki/Complessità_temporale



29. Gioco del Nim

Beat e il suo amico giocano al Nim: 13 fiammiferi sono posti inizialmente su un tavolo e alternativamente i due giocatori ne tolgono 1, 2 o 3. Il giocatore che prende l'ultimo fiammifero è il vincitore.



Aiuto: Se sul tavolo ci sono ancora 4 fiammiferi, il giocatore di turno non può più vincere. Questa situazione deve quindi essere evitata.

Beat inizia, quanti fiammiferi deve togliere per poter vincere il gioco?

- A) 1
- B) 2
- C) 3
- D) Non ha importanza.



Soluzione

La risposta corretta è 1. Sul tavolo restano quindi 12 fiammiferi. L'amico di Beat prende 1, 2 o 3 fiammiferi e al prossimo turno Beat dovrà fare in modo di lasciarne 8 sul tavolo. Di nuovo, l'amico prenderà 1, 2 o 3 fiammiferi e Beat ne lascerà sul tavolo 4. A questo punto Beat ha vinto.

Se Beat dovesse prendere 2 o 3 fiammiferi, il suo amico potrebbe fare in modo di lasciarne 4 sul tavolo, analogamente a quanto descritto sopra. A quel punto Beat non potrebbe più vincere.

Questa è l'informatica!

Nella teoria dei giochi, il gioco del Nim o il famoso gioco del “dilemma del prigioniero” sono utilizzati come modelli per analizzare problemi strategici del mondo reale e trovare delle soluzioni. Nell'economia di mercato, ad esempio, le conoscenze in questo campo servono ad ottimizzare i prezzi di vendita. Sebbene un prezzo più basso possa facilitare le vendite, spesso diminuisce il guadagno per ogni articolo. Al contrario, un prezzo alto aumenta il guadagno per ogni articolo, ma può anche far diminuire le vendite e quindi il profitto totale. Con i modelli della teoria dei giochi si possono simulare le possibili reazioni degli acquirenti al cambiamento dei prezzi. L'importanza della teoria dei giochi nell'economia è testimoniata dai numerosi premi Nobel attribuiti a ricerche in questo campo.

Siti web e parole chiave

gioco del Nim, teoria dei giochi, albero di decisione

- <https://it.wikipedia.org/wiki/Nim>
- <http://it.goobix.com/giochi-online/nim/> (versione alternativa a quella proposta nel quesito)
- https://it.wikipedia.org/wiki/Dilemma_del_prigioniero



30. Numeri di maglia ordinati

Le figure inserite qui sotto mostrano due squadre, ognuna delle quali composta da 15 giocatori. Entrambe le squadre indossano maglie con i numeri stampati sopra. I giocatori della prima squadra sono ordinati per numero, mentre i giocatori della seconda no.

Squadra 1:



Squadra 2:



Come si può verificare nella maniera più rapida quali numeri sono presenti sia nella squadra 1 che nella squadra 2?

- A) Passo in sequenza i numeri della squadra 1. Per ognuno di essi verifico se nella squadra 2 esiste lo stesso numero.
- B) Passo in sequenza i numeri della squadra 2. Per ognuno di essi verifico se nella squadra 1 esiste lo stesso numero.
- C) È uguale da quale squadra iniziare: entrambi i metodi sono ugualmente veloci.
- D) Dapprima bisogna sapere quali numeri non sono in comune.. Sottraggo questo numero a 15 e ottengo il numero cercato.



Soluzione

La risposta corretta è B). In una serie ordinata si può individuare un certo valore più velocemente che in una serie non ordinata. Un computer utilizzerebbe solo $\log_2(n)$ passaggi di ricerca invece di n . Quindi la risposta B) necessita nel caso peggiore $n \cdot \log_2(n)$ passaggi.

Nella risposta A) non si potrebbe scorrere la lista di numeri della squadra 1 velocemente, in quanto dovremmo occuparci di ogni numero individualmente. Quindi la risposta A) necessita di n^2 passaggi nel caso peggiore, molto più di $n \cdot \log_2(n)$ del caso B).

In virtù di quanto scritto sopra, la risposta C) è errata.

La risposta D) non fornisce di norma la risposta cercata e dunque deve essere scartata.

Questa è l'informatica!

In una serie ordinata si possono trovare dei valori più velocemente che in una serie non ordinata. Come cerchi, ad esempio, la maglia numero 9 nella squadra 1? Cerchi il giocatore centrale che ha il numero 17. 9 è più piccolo, quindi guardi a sinistra. Di nuovo, cerchi il giocatore centrale della parte sinistra della squadra, il quale possiede il numero 7. Allora guardi a destra e trovi velocemente il numero 9. Il trucco consiste nel dimezzare ogni volta lo spazio di ricerca, ciò che in informatica è detto "ricerca binaria". Dato che è possibile utilizzare la ricerca binaria solo sulle collezioni di dati ordinate, gli "algoritmi di ordinamento" sono molto importanti.

Nella spiegazione data sopra n si riferisce al numero di dati da ricercare (nel nostro caso 15).

Siti web e parole chiave

ordinamento, ricerca binaria (o dicotomica)

- https://it.wikipedia.org/wiki/Ricerca_dicotomica



31. Cassy, la tartaruga

La tartaruga Cassy vive nelle terre di Reticolandia, in un campo formato da una griglia di 5 celle x 5 celle. Per vivere, si ciba volentieri di insalata. Ogni giorno sul campo crescono nuove piante di insalata, ma Cassy non conosce in quale punto esatto. Per trovarle, la mattina Cassy inizia la ricerca dal centro del campo e segue determinate istruzioni elencate in un blocco.

Fai in modo che Cassy visiti tutte le celle del campo. Scegli le istruzioni sulla sinistra cliccandogli sopra. Puoi utilizzare le istruzioni più di una volta.

- R è una variabile numerica. Quando viene eseguita la prima volta nel blocco di istruzioni, R ha il valore 1, la seconda volta il valore 2, e così via.
- Cassy può lasciare il campo, ma non può uscire da Reticolandia.
- Cliccando su “Verifica”, puoi verificare il tuo programma.



Soluzione

È possibile inserire nel blocco al massimo 4 istruzioni che vengono ripetute 5 volte. Questo significa che il cammino della tartaruga deve seguire una quello di una spirale. Esistono 4 diverse sequenze di istruzioni che Cassy potrebbe seguire per visitare tutte le celle del campo:

- avanza di R celle, gira a sinistra, avanza di R celle, gira a sinistra

Ripeti 5 volte!

avanza di R celle

gira a sinistra

avanza di R celle

gira a sinistra

- avanza di R celle, gira a destra, avanza di R celle, gira a destra

Ripeti 5 volte!

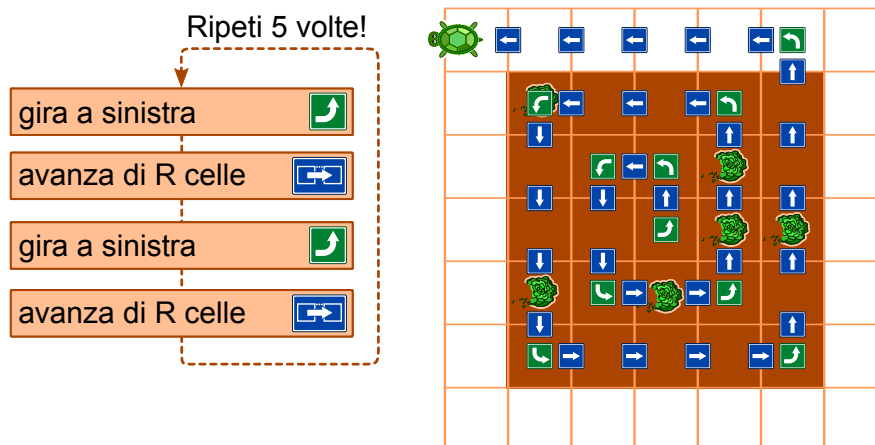
avanza di R celle

gira a destra

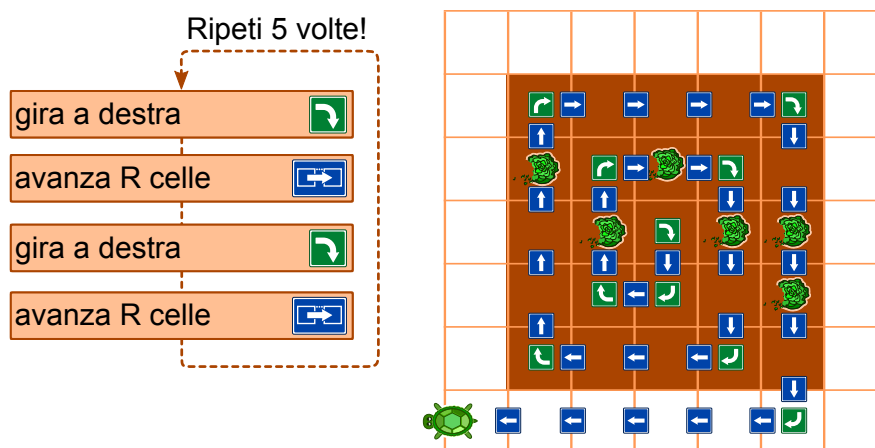
avanza di R celle

gira a destra

- gira a sinistra, avanza di R celle, gira a sinistra, avanza di R celle



- gira a destra, avanza di R celle, gira a destra, avanza di R celle



Questa è l'informatica!

Per questo compito deve essere scritto un vero e proprio programma per computer. Esso è formato da una *sequenza di quattro singole istruzioni*, ripetute 5 volte. Queste ripetizioni sono dette *cicli (o loop)*. Nel nostro caso si tratta di un *count-controlled loop* (ovvero un ciclo controllato da un contatore), dato che R varia da 1 a 5. Un computer esegue queste istruzioni in sequenza. Tutti i linguaggi di programmazione supportano i cicli, oltre ad altre istruzioni che possono controllarne l'esecuzione, come ad esempio la chiamata a sotto-programmi (procedure).

Se il programma è corretto, il computer esegue esattamente ciò che desideri. In caso contrario, il computer eseguirà sempre le istruzioni, ma la tartaruga non si comporterà come ti aspetti. Normalmente, infatti, un computer non è in grado di capire se il risultato di una programmazione è quello voluto oppure no.

Siti web e parole chiave

Turtle-Graphics

- <http://www.turtlegrafik.ch/> (solo inglese o tedesco)
- <http://primaalgo.ch/> (solo tedesco o francese)



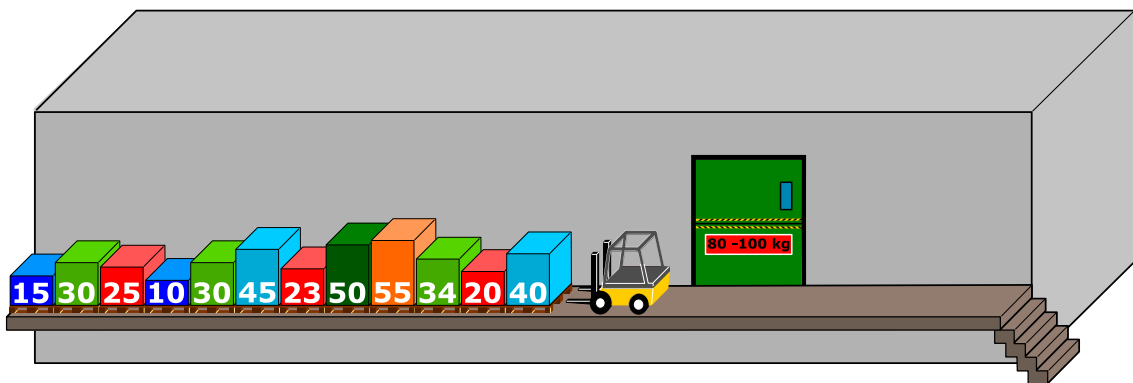
- <http://coesi.inf.usi.ch/eventi-trascorsi> (“Introduzione all’informatica e alla programmazione”)
- <https://xlogo.tuxfamily.org/it/index-it.html>



32. Montacarichi

Su una rampa di carico sono posizionati dei bancali (palette) affiancati. Essi, da sinistra a destra, pesano rispettivamente 15 kg, 30 kg, 25 kg, 10 kg, 30 kg, 45 kg, 23 kg, 50 kg, 55 kg, 34 kg, 20 kg, 40 kg. La rampa è talmente stretta che solo il bancale in cima alla fila può essere preso e spostato dal muletto.

I bancali vengono caricati su un montacarichi che li trasporta fino al magazzino. Il montacarichi entra in funzione non appena il carico raggiunge gli 80 kg. Esso, però, non può sollevare più di 100 kg in totale. Una volta scaricati i bacali, il montacarichi torna alla rampa vuoto.



Come detto, sul montacarichi viene caricato il bancale più vicino nella fila iniziale. Se il peso totale supera i 100 kg, il muletto prende l'ultimo bancale caricato e lo appoggia sull'altro lato della rampa. Altrimenti il bancale rimane sul montacarichi.

Quando tutti i bancali di sinistra sono stati portati al magazzino, il muletto comincia spostare i bancali infilati precedentemente sulla destra, in modo analogo a quanto fatto con la fila di sinistra.

Quale affermazione è corretta tra quelle espote qui di seguito?

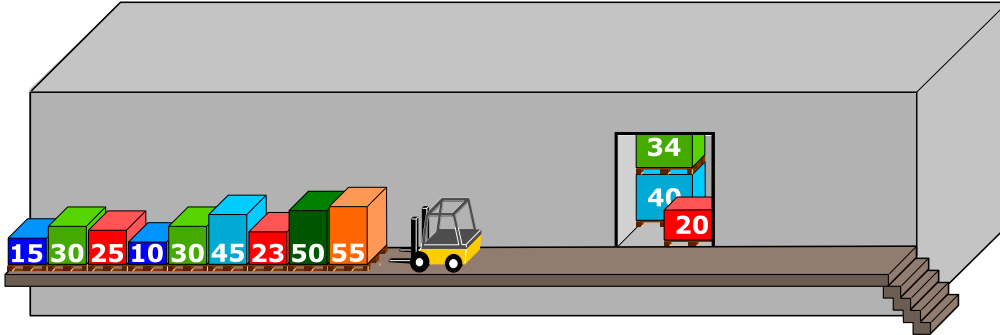
- A) Al secondo viaggio del montacarichi il peso complessivo ammontava a 98 kg.
- B) Non si è mai formata una fila a destra del montacarichi.
- C) In un viaggio del montacarichi il peso complessivo ammontava a 100 kg.
- D) Il montacarichi ha fatto in totale 5 viaggi.
- E) Non è possibile, con la procedura descritta sopra, spostare tutti i bancali nel magazzino.



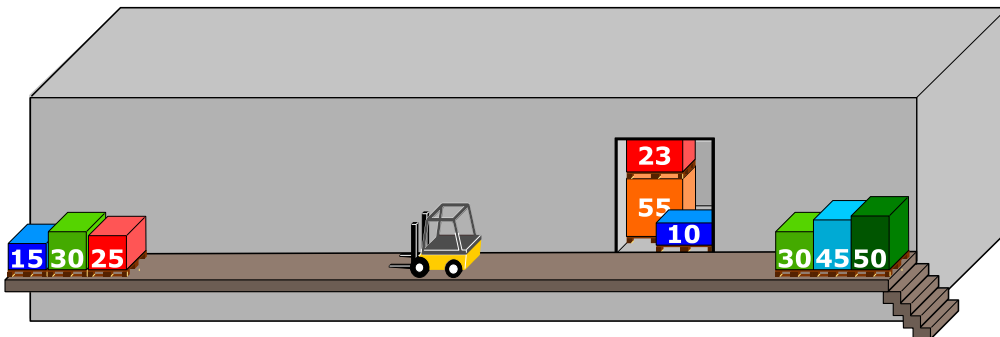
Soluzione

La risposta corretta è C):

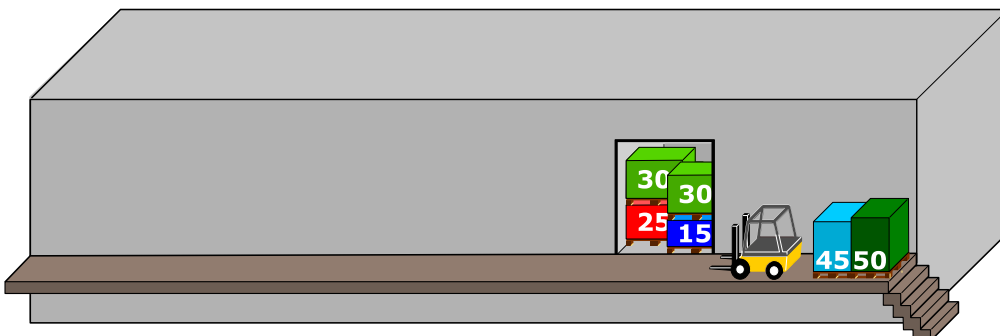
- Il primo carico è composto dai primi tre bancali. Questi pesano: $40\text{ kg} + 20\text{ kg} + 34\text{ kg} = 94\text{ kg}$.



- Il secondo carico è composto inizialmente da un bancale (55 kg). Quando viene caricato il successivo (50 kg), il peso complessivo eccede i 100 kg. Il muletto, quindi, lo toglie mettendolo sulla destra. Il prossimo bancale della fila di sinistra (23 kg) viene caricato e il peso totale raggiunge: $55\text{ kg} + 23\text{ kg} = 78\text{ kg}$. Il bancale successivo (45 kg) risulta di nuovo essere troppo pesante e dunque è spostato sulla fila di destra. La stessa operazione viene compiuta con il bancale che segue (30 kg). Infine, con bancale da 10 kg, il peso totale supera gli 80 kg ($55\text{ kg} + 23\text{ kg} + 10\text{ kg} = 88\text{ kg}$) e il carico può essere trasportato al magazzino.

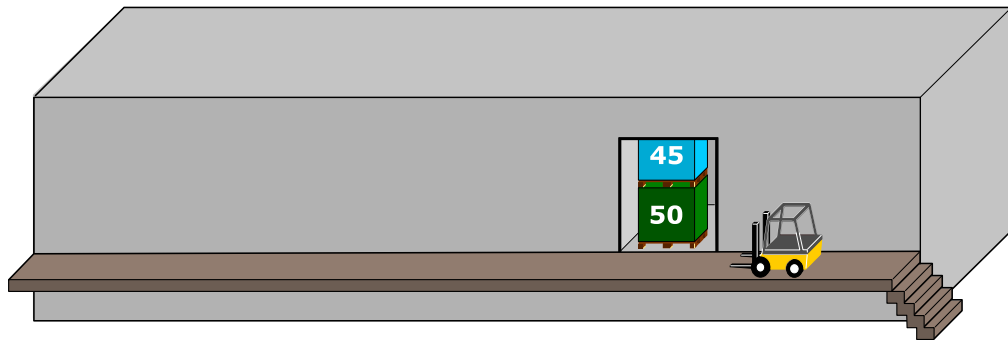


- Gli ultimi 3 bancali ($25\text{ kg} + 30\text{ kg} + 15\text{ kg} = 70\text{ kg}$) vengono caricati sul montacarichi. Dato che il peso è inferiore agli 80 kg, viene aggiunto anche il primo bancale della fila di destra (30 kg), raggiungendo così un totale di 100 kg.





- A questo punto vengono caricati gli ultimi due bancali, per un peso totale di: $45\text{ kg} + 50\text{ kg} = 95\text{ kg}$. Questo è il quarto e ultimo viaggio del montacarichi.



In virtù di questa spiegazione, le risposte A), B), D) ed E) sono chiaramente errate.

Questa è l'informatica!

Siccome la rampa è stretta, solo il primo bancale della fila può essere caricato. Questo comportamento corrisponde a quello di una pila. Quando si impilano in successione degli oggetti solo quello in cima può essere prelevato e, analogamente, solo in cima possono essere aggiunti altri oggetti. Anche in informatica simili strutture di dati vengono dette “pile” (o, più spesso, si usa il termine inglese “stack”). Nel nostro caso vengono formate tre pile: quella di sinistra (con primo elemento tutto a destra), quella di destra (primo elemento a sinistra) e quella nel montacarichi (primo elemento quello più in alto).

Le pile sono abbastanza comuni anche nella vita reale: quando da bambini si costruiscono delle torri con i Lego, solo il mattoncino in cima può essere rimosso senza danni e solo in cima possono venire aggiunti altri mattoncini; anche un cono gelato può essere considerato una pila, dato che l'ultima pallina aggiunta è la prima ad essere mangiata; un ulteriore esempio è la cronologia salvata durante la navigazione sul web: quando si clicca “Indietro” si torna all'ultima pagina visitata.

Il principio alla base di questo comportamento è detto LIFO “last in – first out” (l'ultimo a entrare è il primo a uscire). Esso si contrappone al FIFO “first in – first out” (il primo a entrare è anche il primo a uscire), un principio utilizzato nelle code d'attesa (per esempio dal dottore, anche se per fortuna i casi urgenti hanno la priorità).

Siti web e parole chiave

struttura dei dati, pila (o stack), algoritmo, LIFO, FIFO

- <https://it.wikipedia.org/wiki/FIFO>
- <https://it.wikipedia.org/wiki/LIFO>
- [https://it.wikipedia.org/wiki/Pila_\(informatica\)](https://it.wikipedia.org/wiki/Pila_(informatica))





33. Gioco con le biglie

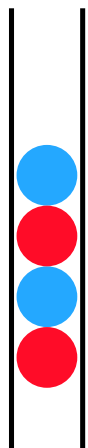
Emil gioca a un nuovo solitario sul suo computer. Il gioco inizia con un minimo di 3 biglie colorate (rosse o blu) impilate in un tubo.

Dopo aver premuto un tasto, le due biglie più in basso cadono fuori dal tubo. Allo stesso tempo, delle nuove biglie vengono impilate dall'alto. A dipendenza del colore della biglia posta inizialmente più in basso, possono accadere due cose:

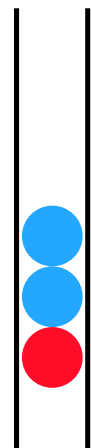
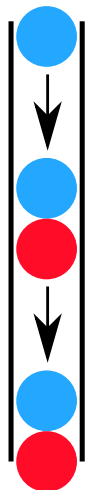
Se la biglia più in basso era rossa, una nuova biglia blu viene impilata in alto:

Se la biglia più in basso era blu, vengono impilate 3 nuove biglie rispettivamente di colore rosso, blu e rosso:

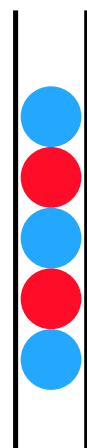
prima



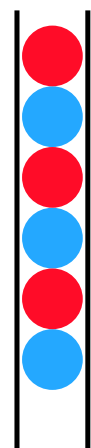
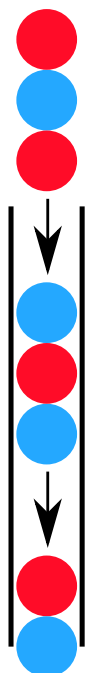
dopo



prima



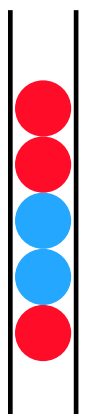
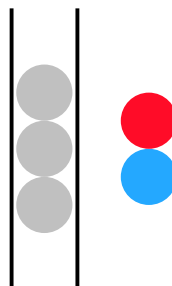
dopo



Fino a quando ci sono almeno 3 biglie nel tubo, Emil continua a premere il tasto. Il gioco termina non appena ci sono meno di 3 biglie nel tubo.

Se inizialmente nel tubo dovessero esserci le biglie mostrate nella figura a destra, dopo cinque pressioni del tasto resterebbero solo 2 biglie blu e il gioco terminerebbe.

Trascina nel tubo le biglie con colori opportuni per fare in modo che il gioco non termini mai.

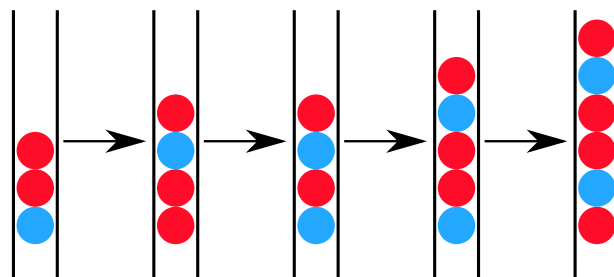
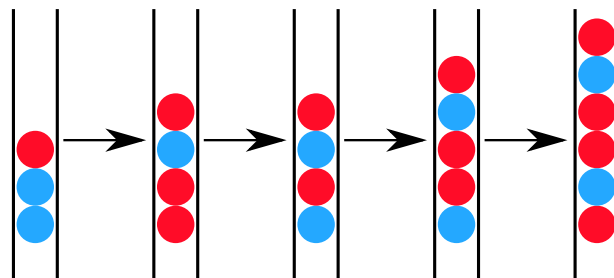
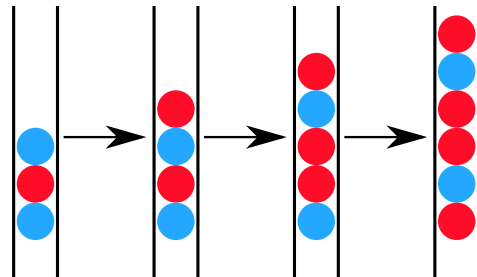
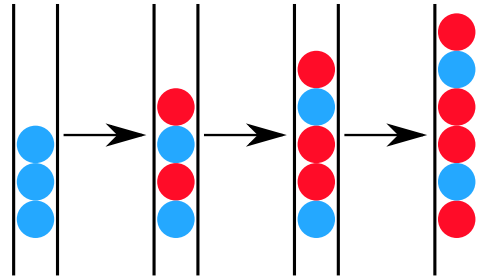




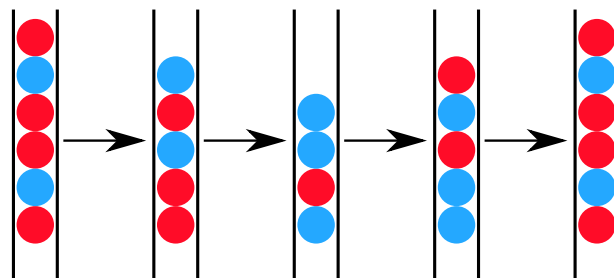
Soluzione

Se la biglia posta più in basso nella pila di 3 biglie fosse rossa, il gioco terminerebbe già dopo la prima pressione del tasto, poiché resterebbero solo 2 biglie.

Al contrario, se l'ultima biglia fosse blu il gioco continuerebbe all'infinito. Con al massimo quattro pressioni del tasto, si otterrebbe la una pila di 6 biglie con colori, risp., rosso-blu-rosso-rosso-blu-rosso:



Con questa configurazione si entrerebbe poi in un ciclo infinito:





Questa è l'informatica!

Il gioco presentato in questo compito fu preso ad esempio da Emil Leon Post per dimostrare come il processo di riscrittura di catene di simboli (stringhe) possa condurre a cicli infiniti (processi senza termine). Emil Leon Post (1897-1954) era un matematico e studioso della logica di origine polacca, che pubblicò numerosi contributi scientifici nel campo dell'informatica teorica e, più precisamente, della logica proposizionale.

Un modello di riscrittura, come quello del nostro esempio, è descritto attraverso una grammatica formale. In essa vengono definite le regole di sostituzione. Nel nostro caso, la grammatica era definita da 4 regole (dove X è una qualsiasi biglia di colore rosso o blu):

$$Xbb \rightarrow rbrX$$

$$Xrb \rightarrow rbrX$$

$$Xbr \rightarrow bX$$

$$Xrr \rightarrow bX$$

Siti web e parole chiave

modelli computazionali, linguaggi formali, regole di produzione, stringhe, sistemi di elaborazione

- http://esolangs.org/wiki/Post_canonical_system (inglese)
- https://en.wikipedia.org/wiki/Tag_system (inglese)
- https://en.wikipedia.org/wiki/Post_canonical_system (inglese)



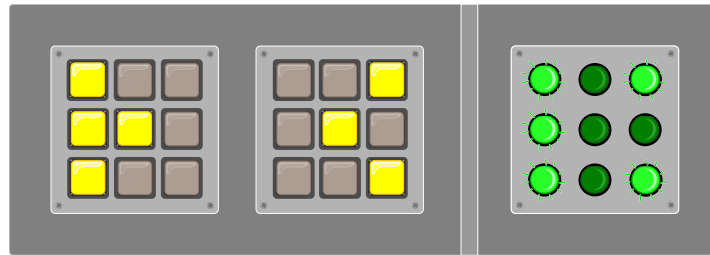


34. Due possibilità

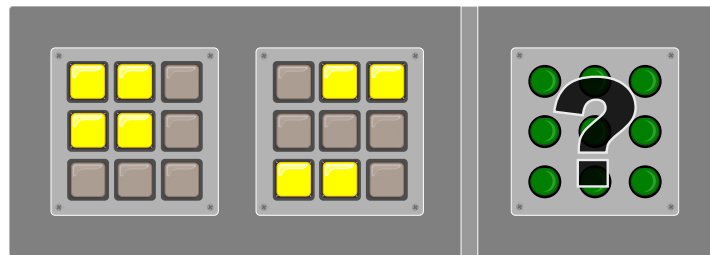
Erich ha trovato un vecchio apparecchio elettronico. Sulla parte sinistra ci sono 2 campi, ognuno dei quali ha 9 tasti. Sulla parte destra c'è invece un campo con 9 luci LED. Premendo i tasti dei campi a sinistra si accendono o si spengono i LED sulla destra.

Erich osserva che la posizione del LED che si accende o spegne è correlata alla stessa posizione e alla combinazione dei tasti dei due campi di sinistra.

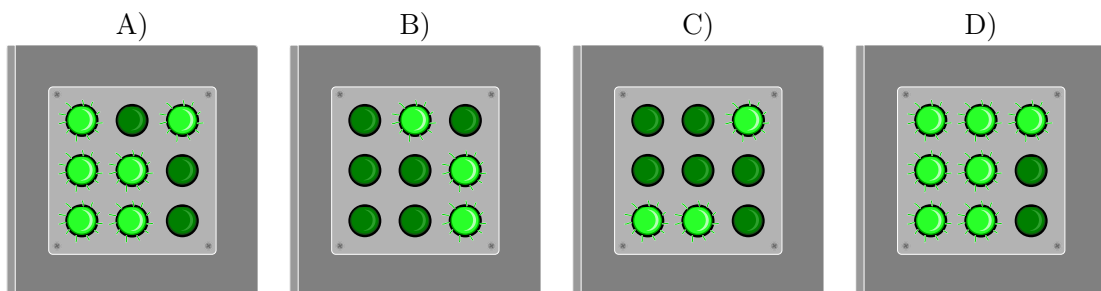
Al momento la combinazione di tasti (sinistra) e i LED accesi (destra) è la seguente:



Erich cambia quindi la combinazione come segue:



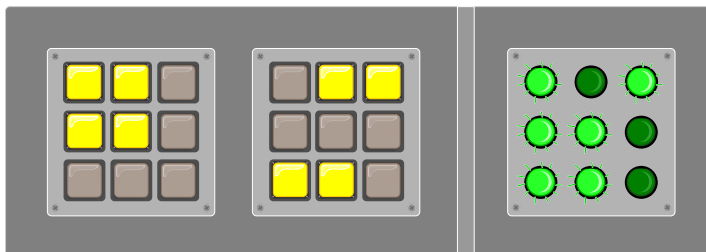
Quali LED si accenderanno sulla destra?





Soluzione

La risposta corretta è A). La regola è: quando i tasti nella stessa posizione nei due campi a sinistra sono entrambi accesi o spenti, allora il LED nella posizione corrispondente a destra è spento. Altrimenti è acceso.



Questa è l'informatica!

In tutti i campi dell'informatica si utilizza spesso la logica binaria per valutare la correttezza o l'inesattezza di determinate asserzioni in una certa circostanza. Questo, naturalmente, con il presupposto che ci siano solo due possibilità di risposta, una terza possibilità è esclusa. Le funzioni logiche, infatti, possono fornire solo due valori: “vero” o “falso”.

Nel nostro compito abbiamo usato la funzione logica XOR (dall'inglese eXclusive OR, “oppure esclusivo”). Essa funziona in questo modo: quando due asserzioni sono entrambe “vere” o entrambe “false” (risp. “accese” o “spente”) allora il risultato è “falso” (“spento”). Negli altri casi (ovvero “falso-vero”, “vero-falso”) essa restituisce un valore “vero”. Nel compito bisognava applicare questa funzione 9 volte (i tasti) e esprimerne il risultato a destra (LED).

Siti web e parole chiave

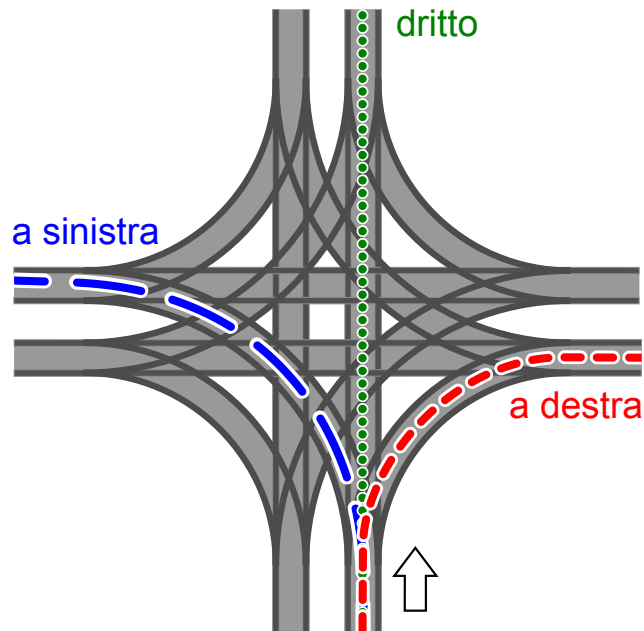
logica binaria, funzione XOR

- https://it.wikipedia.org/wiki/Algebra_di_Boole
- https://it.wikipedia.org/wiki/Disgiunzione_esclusiva
- https://en.wikipedia.org/wiki/XOR_gate (inglese)



35. Incrocio di tram

Sulle strade di St. Pietroburgo circolano dei tram. In città c'è un incrocio che permette di imboccare qualsiasi strada: i tram possono arrivarci da tutte le direzioni e svoltare a destra, a sinistra o continuare dritti.



La direzione in cui vanno i tram è determinata dalla posizione di alcuni scambi. Tale posizione è descritta dalle parole DIRITTO, SINISTRA e DESTRA. Per esempio: la combinazione SINISTRA-DIRITTO-SINISTRA-DESTRA significa che gli scambi sono posti in modo che un tram vada a sinistra, il prossimo (in senso orario) dritto, il successivo (sempre in senso orario) a sinistra e il quarto tram a destra.

Può anche succedere che all'incrocio arrivino tram da tutte le direzioni allo stesso momento.

Quali posizioni degli scambi possono portare a un incidente tra i tram?

- A) DESTRA-DESTRA-DESTRA-DESTRA
- B) DESTRA-DESTRA-SINISTRA-SINISTRA
- C) SINISTRA-DESTRA-SINISTRA-DESTRA
- D) DESTRA-SINISTRA-DESTRA-SINISTRA



Soluzione

La risposta corretta è B). Il primo tram (che svolta a destra) si scontra con il terzo tram (che svolta a sinistra). Analogamente, il secondo tram (che svolta a destra) si scontra con il quarto tram (che svolta a sinistra).

Per ogni altra soluzione, i tram percorrono le rotaie senza alcuna collisione: nella risposta A) vengono percorse tutte le curve “esterne” all’incrocio, nelle risposte C) e D) i tram svoltano sempre lungo la stessa strada in modo parallelo. In queste soluzioni è importante notare che le curve interne non collidono, ciò che viene detto svolta “tangenziale” o “americana” a sinistra.

Questa è l’informatica!

I binari del tram – in particolare in materia di incroci – sono un esempio di risorse condivise da molteplici attori. In un certo momento, solo un tram può trovarsi in un determinato punto . . . altrimenti c’è una collisione. Il coordinamento e l’ottimizzazione della condivisione delle risorse da parte di processi paralleli è un ambito importante dell’informatica. Come con i tram, si deve essere consapevoli dei potenziali rischi e trovare regole per l’accesso sicuro, per esempio, a basi di dati o a periferiche condivise. In caso contrario, potrebbe esserci una perdita di dati o la distruzione di componenti hardware.

Nel caso dell’incrocio è una buona idea consentire solo impostazioni “sicure” degli scambi. Una buona soluzione potrebbe essere anche l’utilizzo di segnali luminosi. Anche nell’informatica si utilizzano dei veri e propri segnali, i cosiddetti “semafori”. Essi indicano se alcune risorse sono già in uso e impongono al processo che vuole accedervi di aspettare il proprio turno.

Siti web e parole chiave

incrocio di tram, sincronizzazione (di processi), accesso a risorse limitate

- <https://it.wikipedia.org/wiki/Sincronizzazione>
- <http://www.swisseduc.ch/informatik/infotraffic/logictraffic/> (tedesco)



36. La codifica delle bandiere

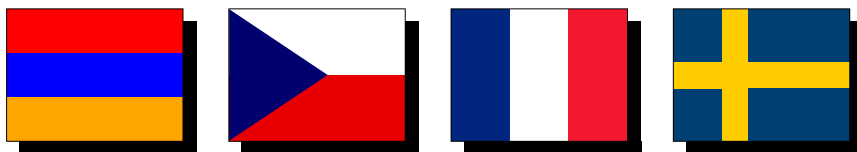
Le immagini del computer sono composte da righe di elementi di immagine (pixel). Nel caso più semplice tali immagini vengono memorizzate come file in cui il colore di ogni pixel è descritto singolarmente. Con il formato per file GIW, le immagini vengono anche compresse e quindi memorizzate in un file di minore dimensione. Il GIW (un formato di pura fantasia) funziona in questo modo:

- Ogni riga di pixel viene descritta singolarmente.
- Ogni colore è descritto attraverso un'abbreviazione di tre lettere.
- Una serie di pixel aventi lo stesso colore è descritta da una coppia di parentesi, contenente il simbolo del colore e il numero di pixel nella serie.

Per esempio, una riga di pixel descritta dalle parentesi (ver, 20) (bia, 13) è formata da 20 pixel verdi e 13 pixel bianchi.

Qui sotto potete osservare quattro immagini di bandiere. Tutte le immagini possiedono lo stesso numero di righe, ognuna delle quali possiede lo stesso numero di pixel. Tali immagini vengono memorizzate in formato GIW.

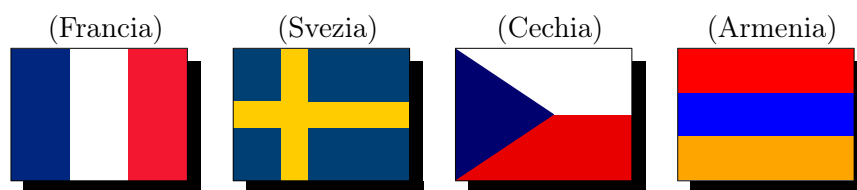
Ordina le immagini per grandezza del file GIW. Trascina le bandiere in modo che a sinistra ci sia la bandiera con il file GIW più grande.





Soluzione

La successione di bandiere esatta è:



Se una riga possiede lo stesso colore, viene descritta da una singola coppia di parentesi. Per ogni successivo cambio di colore all'interno di una riga si aggiunge poi una coppia di parentesi nella codifica. Per esempio, la bandiera dell'Armenia ha lo stesso colore in ogni riga e quindi ogni riga sarà codificata con una singola coppia di parentesi.

La bandiera della Cechia, invece, cambia colore 1 volta per ogni riga: dapprima passa dal blu al bianco e poi dal blu al rosso. Il file GIW ha dunque bisogno di 2 coppie di parentesi per ogni riga. La bandiera francese è rappresentata da 3 serie di colori per ogni riga e dunque necessita di 3 coppie di parentesi.

La bandiera svedese ha una striscia gialla orizzontale e per questa zona ha bisogno di una sola coppia di parentesi per ogni riga. Nel resto della bandiera il colore cambia 2 volte, ci vogliono quindi 3 coppie di parentesi per codificare ogni riga. Da questo si deduce facilmente che la bandiera svedese utilizza più parentesi di quella armena e meno di quella francese.

Non ci resta quindi che confrontare la bandiera svedese con quella ceca. Se la bandiera svedese avesse esattamente lo stesso numero di righe nella striscia gialla orizzontale come nella parte rimanente, avremmo bisogno in media di 2 coppie di parentesi per ogni riga. La striscia orizzontale è però chiaramente più piccola del resto della bandiera, dunque possiamo concludere che in media abbiamo bisogno più di 2 coppie di parentesi per ogni linea e affermare quindi che la bandiera svedese necessita di un file GIW più grande di quella ceca.

Questa è l'informatica!

La compressione dei dati rappresenta un ambito importante dell'informatica. Grazie alla riduzione dello spazio di memoria necessario, si possono trasportare i dati digitali attraverso la rete senza perdite di tempo. Gli algoritmi di compressione dei dati possono velocizzare notevolmente la comunicazione. Se, ad esempio, si ascoltasse la musica di una web radio senza compressione dei dati, la larghezza di banda necessaria sarebbe anche 10 volte superiore. Per questo la ricerca di nuovi algoritmi di compressione è intensa, così da poter salvare foto, musica o video in modo sempre più efficiente. La compressione utilizzata nel nostro quesito appartiene al tipo RLE (Run-length encoding). Ulteriori informazioni in merito possono essere visualizzate nel seguente filmato YouTube: https://www.youtube.com/watch?v=ydpNscvym_E.

Siti web e parole chiave

codifica, compressione, algoritmi di compressione, grafica bitmap

- https://it.wikipedia.org/wiki/Grafica_raster
- https://it.wikipedia.org/wiki/Compressione_dei_dati
- https://it.wikipedia.org/wiki/Run-length_encoding
- https://www.youtube.com/watch?v=ydpNscvym_E



37. Lacci da scarpe

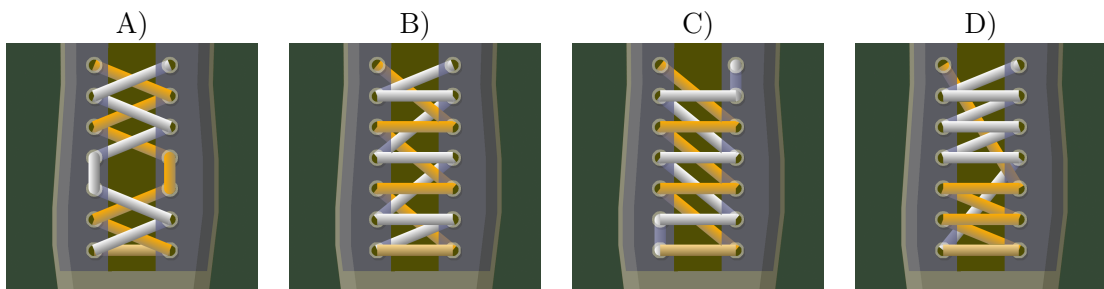
Ai castori piace creare delle allacciature della scarpa alla moda. A questo scopo, possono programmare una macchina che lega i lacci da scarpe per loro. Essa esegue i seguenti comandi:

Comando	Significato
{ ... }	Il contenuto delle parentesi viene ripetuto il più possibile.
n { ... }	Il contenuto delle parentesi viene ripetuto n volte.
arancio:	I comandi che seguono valgono solo per i lacci arancioni.
bianco:	I comandi che seguono valgono solo per i lacci bianchi.
esterno:	Il laccio passa attraverso l'occhiello, dall'interno verso l'esterno della scarpa.
interno:	Il laccio passa attraverso l'occhiello, dall'esterno verso l'interno della scarpa.
sopra:	Il laccio viene spostato verso l'alto (al prossimo occhiello).
sotto:	Il laccio viene spostato verso il basso (all'occhiello che precede).
cambio:	Il laccio viene spostato dall'occhiello di destra a quello di sinistra o viceversa.

Quale allacciatura crea il seguente programma?:

```

arancio: esterno
bianco: esterno
2{
    arancio: sopra cambio esterno
    bianco: sopra cambio esterno
}
arancio: sopra interno
bianco: sopra interno
{
    arancio: sopra cambio esterno
    bianco: sopra cambio esterno
}
    
```





Soluzione

La soluzione corretta è A). Per i primi 2 occhielli (dal basso) si applica il contenuto delle prime parentesi 2 volte, i lacci vengono poi spostati verso l'alto all'occhiello successivo. In seguito si ripetono i comandi contenuti nell'ultima parentesi fino alla fine.

Questa è l'informatica!

La macchina che crea le allacciature viene programmata con un linguaggio di programmazione semplice. Esso contiene i tipici elementi che compongono i linguaggi usati dagli informatici:

- Esecuzione sequenziale dei comandi (*Sequenze di istruzioni*)
- Ripetizione di blocchi di comandi (*cicli o loop*) e ripetizione condizionata o limitata

Ciò che manca nel linguaggio utilizzato in questo esercizio sono le *istruzioni condizionali* (selezioni), che invece vengono spesso utilizzate nella pratica, così come la creazione separata di blocchi di istruzioni funzionali (*modularizzazione*).

La macchina per le allacciature può essere anche programmata in un simulatore: <http://www.klbg.com/lacing/>. In questo caso, però, i comandi utilizzano vocaboli inglesi.

Siti web e parole chiave

cicli, comandi o istruzioni, simulatore, lacci da scarpe

- [https://it.wikipedia.org/wiki/Interprete_\(informatica\)](https://it.wikipedia.org/wiki/Interprete_(informatica))
- https://it.wikipedia.org/wiki/Linguaggio_di_programmazione
- <http://www.klbg.com/lacing/>
- <http://www.fiegggen.com/shoelace/lacingmethods.htm>

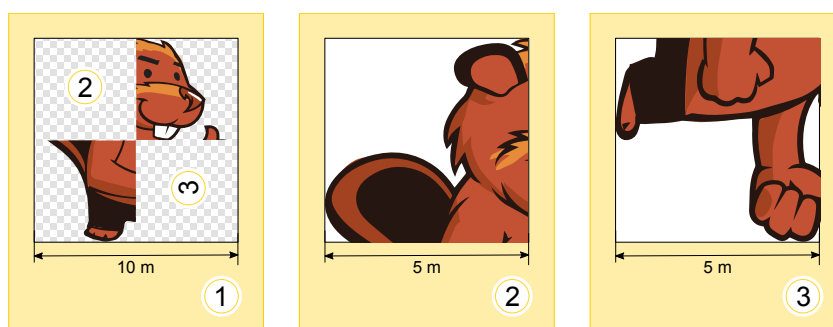


38. Pittura ricorsiva

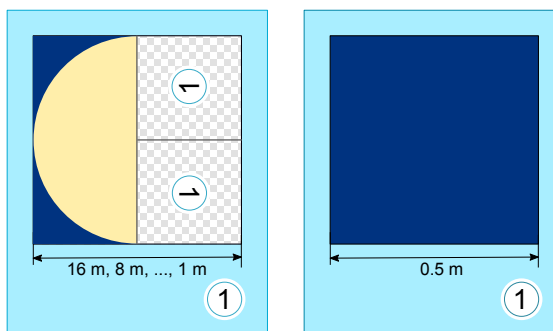
Tina e Ben aiutano a preparare una mostra nel museo di informatica. Sul pavimento di una sala d'esposizione dovrebbero dipingere un quadro di dimensioni 16 metri \times 16 metri.

Da alcuni artisti ottengono un set di schede da pittura speciali. Esse possono creare un quadro grazie a istruzioni del famoso "linguaggio dei pittori": queste istruzioni particolari contengono informazioni sugli elementi del disegno, sulle misure e sulle rotazioni. Su alcune schede da pittura sono presenti dei campi numerati che fanno riferimento ad altre schede.

Ecco un esempio tratto da un precedente progetto fatto con queste schede speciali. Se le istruzioni vengono eseguite correttamente, si ottiene il dipinto di un castoro. Si noti che le schede hanno lunghezze dei bordi diverse.

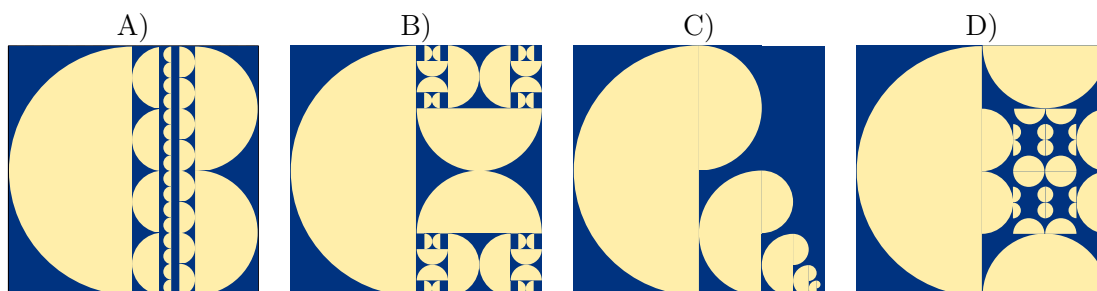


Per la sala d'esposizione Tina e Ben ricevono le seguenti schede da pittura:



Ben è confuso: "Come è possibile? La scheda a sinistra si riferisce a sé stessa, e poi le due schede hanno lo stesso numero!". Ridendo Tina dice: "Ce la faremo! Utilizziamo dapprima solo la scheda di sinistra. Useremo quella di destra in un secondo tempo, come istruzione per terminare il dipinto."

Come apparirà il pavimento della sala d'esposizione?





Soluzione

La scheda di sinistra indica che la metà sinistra del pavimento deve essere riempita con un semi-cerchio con il lato tondo orientato a sinistra. Per la metà destra, si deve utilizzare la stessa scheda due volte, ma con dimensioni ridotte. L'orientamento degli elementi dell'immagine deve essere conforme all'orientamento indicato dai campi numerati.

In tale scheda, il campo numerato posto in alto è ruotato di 90° a sinistra. Pertanto, il semi-cerchio deve essere ruotato verso sinistra e quindi il lato tondo apparirà orientato verso il basso.

In campo numerato posto in basso è invece ruotato di 90° verso destra. Pertanto, il semi-cerchio deve essere ruotato verso destra e quindi il lato tondo apparirà orientato verso l'alto.

Questa situazione si verifica solo nella risposta B). Quindi B) è l'unica risposta corretta.

Questa è l'informatica!

In informatica, gli algoritmi (ovvero insiemi di istruzioni) che richiamano sé stessi sono detti "ricorsivi". Il termine deriva dal latino "recurrere" (qui nel senso di "ritorno a sé stesso"). La ricorsione è un uno strumento potente: alcuni compiti complessi possono essere risolti in modo semplice con l'utilizzo di algoritmi ricorsivi. Alcune procedure ricorsive, tuttavia, sono molto difficili da capire.

Un algoritmo ricorsivo contiene sempre una "chiamata" a sé stesso e una condizione di terminazione, che si verifica quando la ricorsione deve terminare. Se questa condizione non si verificasse, la ricorsione continuerebbe all'infinito o, almeno, fino all'esaurimento di una qualsiasi risorsa (come, ad esempio, la memoria).

Siti web e parole chiave

programmazione, ricorsione, condizione di terminazione

- https://it.wikipedia.org/wiki/Algoritmo_ricorsivo



39. Giocando con le forme

Alicia si diverte con un gioco basato sulle forme geometriche. In esso, crea delle sequenze di forme in passi successivi, utilizzando ogni volta una serie di regole di sostituzione. Un esempio di regole di sostituzione è:



Alicia parte sempre da una singola forma. Ad esempio, iniziando con un quadrato e utilizzando le regole di sostituzione sopra descritte, i primi 3 passi sarebbero:



In un'ulteriore sessione di gioco, Alicia ottiene la seguente sequenza di forme utilizzando altre regole:



Quali regole di sostituzione ha utilizzato in questa sessione di gioco?

A)	B)	C)	D)
$\square \rightarrow \triangle \square \square$	$\square \rightarrow \square \circ \circ$	$\triangle \rightarrow \triangle \triangle$	$\triangle \rightarrow \square \circ$
$\triangle \rightarrow \circ$	$\triangle \rightarrow \triangle \square$	$\square \rightarrow \circ \circ$	$\square \rightarrow \triangle \triangle \triangle$
$\circ \rightarrow \triangle \triangle$	$\circ \rightarrow \square \triangle$	$\circ \rightarrow \triangle \square \triangle$	$\circ \rightarrow \square \triangle$



Soluzione

La risposta corretta è B). Iniziando con un triangolo, Alicia esegue le sostituzioni seguenti:



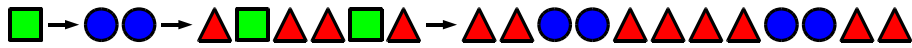
Le altre possibilità vengono scartate poiché:

Risposta A). Se Alicia iniziasse con un triangolo o un cerchio non potrebbe mai ottenere un quadrato. Iniziando con un quadrato, invece, otterrebbe:



Dato che ogni serie cresce in lunghezza ad ogni passo, da questa situazione è impossibile ottenere la sequenza desiderata;

Risposta C). Se Alicia iniziasse con un triangolo, non potrebbe mai ottenere né un quadrato né un cerchio. Iniziando con un quadrato, invece, otterrebbe:



e i due triangoli iniziali non possono essere trasformati in un quadrato, come nella sequenza desiderata. Infine, iniziando con un cerchio, otterrebbe:



e anche in questo caso il problema dei due triangoli iniziali non potrebbe essere risolto; Risposta D). Non sarebbe possibile creare 2 cerchi adiacenti. In particolare, non sarebbe possibile creare più di un cerchio, dato che ogni cerchio, nel passo successivo, sarebbe sostituito da un quadrato e un triangolo. Per questo non si può ottenere la combinazione desiderata.

Questa è l'informatica!

Le regole utilizzate nel nostro esempio sono di fatto un insieme di regole di riscrittura (“string rewriting system (SRS)”) che descrivono in concreto una grammatica context-free, in cui ogni parola (qui “forma geometrica”) viene sostituita da una o più parole.

Una grammatica context-free è formata da:

- un insieme di parole o simboli (nel nostro caso forme geometriche , e). Contrariamente al nostro esempio, esistono due tipologie di parole: quelle che possono essere ulteriormente sostituite (“non terminali”) e altre che non possono esserlo più (“terminali”).
- un simbolo di inizio (nel nostro caso si può scegliere tra le forme geometriche disponibili)
- un insieme di regole di produzione (nel nostro caso le regole di sostituzione)

Contrariamente al nostro esempio, nelle grammatiche context-free non sempre vengono utilizzate tutte le regole di produzione contemporaneamente, si può invece scegliere quale regola applicare.

Le grammatiche context-free vengono utilizzate in vari contesti: da linguaggi formali, come quelli di programmazione, allo studio di lingue naturali (italiano, tedesco, ...) fino a giungere alla descrizione del processo di crescita nelle piante.

Per risolvere questo quesito, oltre a capire le regole di sostituzione, si deve anche possedere la capacità di analizzare le sequenze di simboli. Quest'ultima operazione in informatica è detta “parsing” ed è un passo fondamentale per trasformare le istruzioni scritte in un programma in comandi che il computer può comprendere ed eseguire.



Siti web e parole chiave

grammatica context-free (CFG), parsing

- https://it.wikipedia.org/wiki/Grammatica_libera_dal_contesto
- <https://en.wikipedia.org/wiki/L-system> (in inglese)
- <https://it.wikipedia.org/wiki/Parsing>





40. Scoiattoli egoisti

Gli scoiattoli vivono nelle buche degli alberi. Un albero ha cinque buche poste una sopra all'altra. Su di esso vivono sedici scoiattoli, suddivisi nelle cinque buche.

Tutte le mattine, ogni scoiattolo controlla in quale buca ci siano stati meno scoiattoli. Più precisamente, oltre che nella propria, esso controlla quanti vicini avrebbe nella buca immediatamente sopra e quanti nella buca immediatamente sotto a quella in cui vive. Per la notte successiva sceglie quindi la buca in cui avrebbe meno scoiattoli vicini. Se nelle buche prese in esame dovesse esserci lo stesso numero di vicini, esso preferirà la propria buca alle altre e la buca superiore a quella inferiore.

Per esempio, se nella notte appena trascorsa nelle buche si trovassero dall'alto in basso 5, 0, 0, 4 e 7 scoiattoli, il giorno successivo la situazione sarebbe la seguente: tutti gli scoiattoli della buca superiore si trasferiscono in quella immediatamente sotto (0 vicini sono meglio di 4); i 7 scoiattoli della buca inferiore si trasferiscono nella buca immediatamente sopra (4 vicini sono meglio di 6); infine, i 4 scoiattoli restanti si trasferiscono nella buca immediatamente sopra (0 vicini sono meglio di 3).

Se, oggi, nelle buche dall'alto verso il basso dovessero esserci 6, 3, 3, 0 e 4 scoiattoli, in quanti giorni si ritroverebbero tutti nella stessa buca?

- A) In due giorni.
- B) In tre giorni.
- C) In quattro giorni.
- D) Non saranno mai tutti nella stessa buca.





Soluzione

La soluzione corretta è: in tre giorni.

Oggi:	6, 3, 3, 0, 4
Dopo 1 giorno:	0, 9, 0, 7, 0
Dopo 2 giorni:	9, 0, 7, 0, 0
Dopo 3 giorni:	0, 16, 0, 0, 0

Questa è l'informatica!

Questo problema è un classico esempio della swarm-intelligence (“intelligenza dello sciame”). L'idea alla base di questo algoritmo è che si possano risolvere problemi complessi attraverso l'unione di elementi molto semplici, sempre che se ne disponga a sufficienza. Per esempio, le formiche si comportano secondo semplici regole, indipendentemente una dall'altra. Quando ce ne sono molte, però, esse sono in grado di svolgere compiti impegnativi, come ad esempio costruire un formicaio e le sue gallerie, trovare la via ottimale verso una risorsa o tritare delle foglie intere.

Anche nel nostro caso abbiamo una certa quantità di elementi (gli scoiattoli) che si comportano secondo semplici regole. Il loro comportamento collettivo è però ben lungi dall'essere “intelligente”. Essi desiderano avere più spazio possibile, ma alla fine si ritroveranno tutti nella stessa buca. Da ciò ne consegue che il comportamento di uno sciame non può essere trasformato – uno a uno – in un algoritmo: a volte è infatti meglio cooperare che comportarsi egoisticamente.

Siti web e parole chiave

swarm intelligente (“intelligenza dello sciame”), algoritmi ant-colony (“colonia di formiche”)

- https://it.wikipedia.org/wiki/Swarm_intelligence
- https://it.wikipedia.org/wiki/Intelligenza_collettiva
- https://it.wikipedia.org/wiki/Algoritmo_delle_colonie_di_formiche



A. Autori dei quesiti

 Nursultan Akhmetov	 Hans-Werner Hein	 Erkulan Nurtazanov
 Rosa Alexos	 Mathias Hiron	 Henry Ong
 Guðjón Karl Arnarson	 Sarah Hobson	 Serena Pedrocchi
 Wilfried Baumann	 Martin Horvath	 Wolfgang Pohl
 Daphne Blokhuis	 Juraj Hromkovič	 Ilya Posov
 Ivo Blöchliger	 Yukio Idosaka	 Sergei Pozdniakov
 Andrea Brabcová	 Mile Jovanov	 Dániel Pressing
 Eugenio Bravo	 Martina Kabátová	 J. P. Pretti
 Nicolas Brunner	 Filiz Kalelioğlu	 Lorenzo Repetto
 Alexander Cirri	 Joseph Kaperst	 Kirsten Schlüter
 Valentina Dagienė	 Akiko Kikui	 Eljakim Schrijvers
 Christian Datzko	 Ries Kock	 Sue Sentance
 Susanne Datzko	 Ágnes Kocsis	 Maiko Shimabuku
 Janez Demšar	 Tobias Kohn	 Emil Stankov
 Marissa Engels	 Ivana Kosírová	 Björn Steffen
 Olivier Ens	 Bernd Kurzmann	 Gabrielė Stupurienė
 Hanspeter Erni	 Greg Lee	 Seiichi Tani
 Jürgen Frühwirth	 Dan Lessner	 Peter Tomcsányi
 Gerald Futschek	 Hiroki Manabe	 Monika Tomcsányiová
 Peter Garscha	 Khairul A. Mohamad Zaki	 Willem van der Vegt
 Haris Gavranovic	 Hamed Mohebbi	 Jiří Vaníček
 Yasemin Gülbahar	 Nataša Mori	 Troy Vasiga
 Martin Guggisberg	 Anna Morpurgo	 Lina Vinikienė
 Urs Hauser	 Tom Naughton	 Michael Weigend



B. Sponsoring: concorso 2016

HASLERSTIFTUNG

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

ROBOROBO

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

d digitec.ch

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

**bischof
berger**

<http://www.baerli-biber.ch/>

verkehrshaus.ch

<http://www.verkehrshaus.ch/>
Museo Svizzero dei Trasporti

 **Kanton Zürich
Volkswirtschaftsdirektion
Amt für Wirtschaft und Arbeit**

Standortförderung beim Amt für Wirtschaft und Arbeit
Kanton Zürich

Information **plus Automatik** **Chunsch druus?**
Das ergibt Informatik.

i-factory (Museo Svizzero dei Trasporti, Lucerna)

 **UBS**

<http://www.ubs.com/>
Wealth Management IT and UBS Switzerland IT

bbv
Software Services

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

PRESENTEX
Das Geschenk - die gute Werbung

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



ITgirls@hslu

<https://www.hslu.ch/de-ch/informatik/agenda/veranstaltungen/fuer-schulen/itgirls/>
HLSU, Lucerne University of Applied Sciences and Arts
Engineering & Architecture

PH LUZERN
PÄDAGOGISCHE
HOCHSCHULE

<http://www.phlu.ch/>
Pädagogische Hochschule Luzern

ABZ

AUSBILDUNGS- UND BERATUNGSZENTRUM
FÜR INFORMATIKUNTERRICHT

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



C. Ulteriori offerte

010100110101011001001001
010000010010110101010011
010100110100100101000101
001011010101001101010011
010010010100100100100001

SS!

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

Diventate membri della SSII <http://svia-ssie-ssii.ch/verein/mitgliedschaft/> sostenendo in questo modo il Castoro Informatico.

Chi insegna presso una scuola dell'obbligo, media superiore, professionale o universitaria in Svizzera può diventare membro ordinario della SSII.

Scuole, associazioni o altre organizzazioni possono essere ammesse come membro collettivo.