Activiteit 17

De Peruviaanse toss—Cryptografische protocollen

Samenvatting

Deze activiteit toont hoe je een eenvoudige, maar toch schijnbaar onmogelijke taak kunt volbrengen: een eerlijke willekeurige keuze door het opgooien van een munt, door twee mensen die elkaar niet noodzakelijkerwijs vertrouwen en alleen met elkaar verbonden zijn per telefoon.

Leerplan link

Technologie Niveau 1: Technologische systemen. Begrijpen dat technologische systemen inputs, gecontroleerde veranderingen en outputs hebben.

Vaardigheden

- Booleaanse logica,
- Functies,
- Puzzels oplossen.

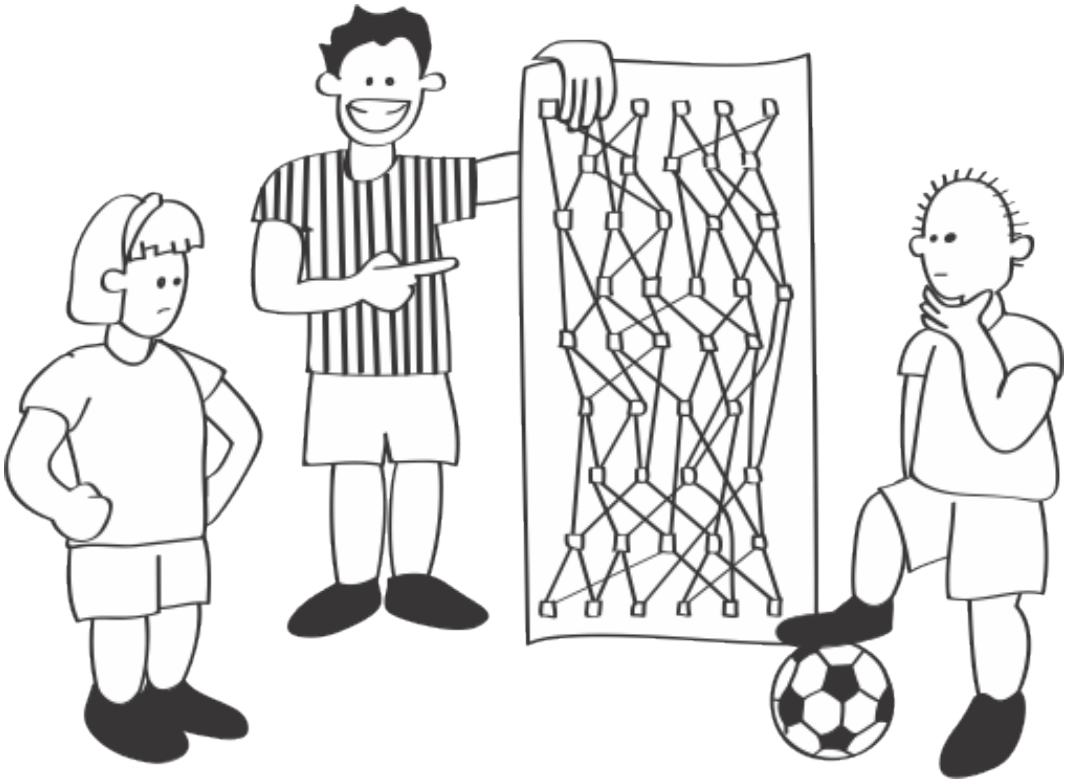
Leeftijd

9 jaar en ouder

Materialen

Voor elke groep leerlingen:

- een kopie van het werkblad De Peruviaanse toss
- ongeveer 2 dozijn kleine knopen of fiches van 2 verschillende kleuren



De Peruviaanse toss

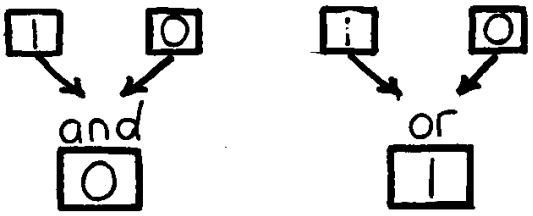
Kinderen leren meer van deze activiteit als ze de binaire nummer representatie kennen (zie Activiteit 1, Tel de punten), het concept van pariteit (zie Activiteit 4, Kaarten truc) hebben geleerd, en het voorbeeld van een-richtings-functies hebben gezien (zie Activiteit 14, Toeristenstad.)

Introductie

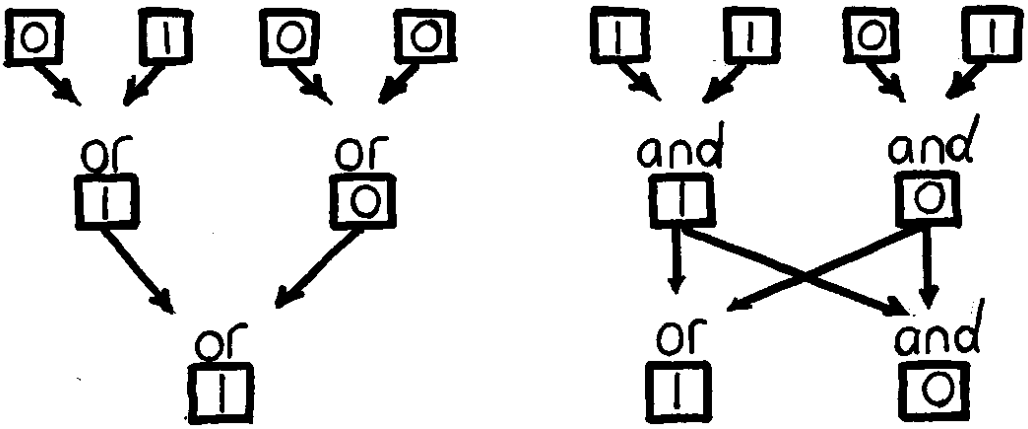
Deze activiteit werd bedacht toen één van de auteurs (MRF) werkte met leerlingen in Peru, vandaar de naam. Je kunt het verhaal aanpassen aan de lokale omstandigheden. De voetbalteams van Lima en Cuzco moeten beslissen wie de thuisploeg wordt voor het kampioenschap. De simpelste manier is een munt opgooien. Maar de steden liggen ver uit elkaar en Alicia, die Lima vertegenwoordigt, en Benito, die Cuzco vertegenwoordigt, hebben geen tijd en geen geld om bij elkaar te komen om een munt op te gooien. Kunnen ze het doen over de telefoon? Alicia kan gooien en Benito kan zeggen kop of munt. Maar dat gaat niet werken, want als Benito "kop" zegt, dan kan Alicia gewoon zeggen "sorry, het was munt" en Benito zou niet weten of dat zo was. Alicia is geen echte bedriegster maar dit is wel een belangrijke wedstrijd en de verleiding is erg sterk. Zelfs als Alicia eerlijk zou blijven, zou Benito dan geloven dat hij echt heeft verloren?
Dit is wat ze beslissen. Samen ontwerpen ze een circuit van and-poorten en or-poorten,

(en-poorten en of-poorten) zoals hier beneden uitgelegd. In principe kunnen ze dit per telefoon doen maar in de praktijk zou dat best saai kunnen worden. (email zou ook kunnen werken!) Tijdens het maken willen ze allebei de zekerheid dat het circuit complex genoeg is zodat de ander niet kan vals spelen. Het laatste circuit is algemeen bekend.

Discussie



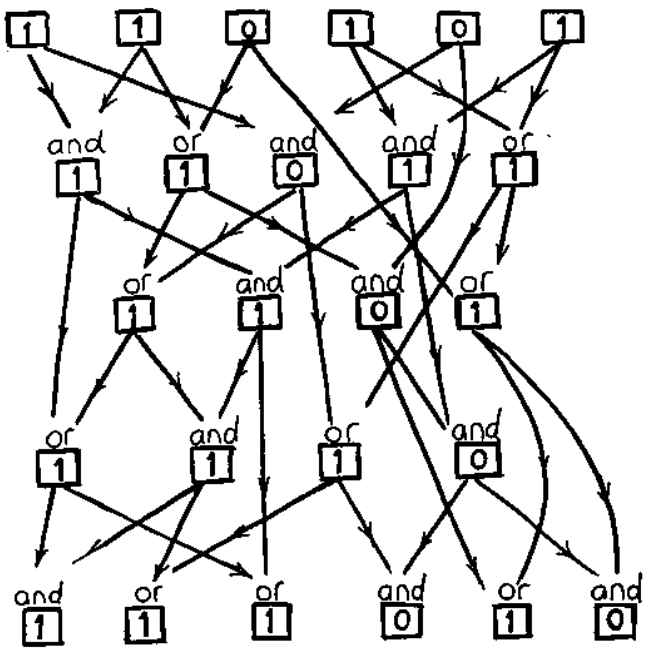
De regels van de and-poorten en or-poorten zijn simpel. Iedere "poort" heeft twee inputs en één output. Elke van de inputs kan een 0 of een 1 zijn, die je ook zou kunnen lezen als respectievelijk false en true, (onwaar en waar). De output van een and-poort is alleen één (true) als beide inputs één zijn en nul (false) in elk ander geval. Bijvoorbeeld hierboven, de and-poort heeft een één en een nul als inputs (bovenste rij) dus is de output (het vierkant onderaan) een nul. De output van een or-poort is één (true) als een van beide (of allebei) inputs een één is, alleen als beide inputs nul zijn is de output nul (false). Dus is de output van de or-poort een één in dit geval, een input van een nul en een één.



De output van een poort kan verbonden worden met de input van een andere poort (of meerdere poorten) om een ingewikkelder effect te genereren. Bijvoorbeeld in het linker circuit zijn de outputs van twee or-poorten verbonden met de input van een derde or-poort met het effect dat als er maar één van de inputs in het circuit een één is de output ook een één zal zijn. In het rechter circuit vormen de outputs van de twee bovenste and-poorten de input in de twee poorten daaronder dus komen er uit dit circuit twee waardes.

Voor de Peruviaanse toss hebben we zelfs nog ingewikkeldere circuits nodig. Het circuit op het werkblad heeft zes inputs en zes outputs. Hier is een uitgewerkt voorbeeld voor een bepaalde set van input waarden.

De manier waarop dit circuit gebruikt kan worden bij het opgooien van een munt via de telefoon is als volgt: Alicia kiest een willekeurige input voor het circuit dat bestaat uit zes binaire getallen (nullen en enen), die zij geheim houdt. Zij stopt de zes cijfers in het circuit en stuurt Benito de zes bits uit de output. Als Benito de output ontvangen heeft moet hij raden of Alicia's input een even of oneven aantal enen bevat - met andere woorden hij moet de parity (pariteit) van Alicia's input raden. Als het circuit ingewikkeld genoeg is kan Benito de uitkomst niet achterhalen en zal zijn antwoord een gok zijn (hij zou daarvoor zelfs een muntje op kunnen gooien om tot zijn antwoord te komen!) Benito wint - en de wedstrijd zal in Cuzco zijn - als zijn gok correct is. Alicia wint - en de wedstrijd is in Lima - als Benito verkeerd gokt. Als Benito Alicia zijn keuze (even of oneven) heeft doorgegeven zal Alicia haar geheime input onthullen zodat Benito kan controleren of deze input de output geeft die ze had doorgegeven.



Verdeel de leerlingen in kleine groepjes en geef iedere groep het circuit, een aantal fiches of iets dergelijks in twee kleuren en leg vervolgens het verhaal uit. De situatie zal wat meer gaan leven voor de leerlingen als ze zich voorstellen de trainer van hun favoriete sportclub te zijn en dat ze een toss moeten organiseren met een rivaliserende club in een andere stad. Stel een legenda op voor de fiches - bijvoorbeeld rood is 0 en blauw is 1- en laat de leerlingen dit bovenaan hun werkblad zetten om het te onthouden. Met de fiches kan je het werkblad meerdere keren gebruiken.

Laat de leerlingen zien hoe je de fiches op de inputs legt om de getallen te laten zien die Alicia had gekozen. Leg vervolgens de regels van de and-poorten en or-poorten uit. Deze worden onderaan het werkblad nog even kort uitgelegd (leerlingen kunnen deze eventueel inkleuren met behulp van de gekozen legenda).

Laat zien hoe je door het circuit gaat, door fiches op de kruispunten te leggen om tot de bijbehorende output te komen. Dit moet heel precies en voorzichtig gebeuren. Fouten

haal je er niet zo maar uit. De onderstaande tabel (niet aan de leerlingen laten zien) laat elke output zien bij alle mogelijke inputs voor je eigen referentie als je twijfelt.

Input	000000	000001	000010	000011	000100	000101	000110	000111
Output	000000	010010	000000	010010	010010	010010	010010	010010
Input	001000	001001	001010	001011	001100	001101	001110	001111
Output	001010	011010	001010	011010	011010	011010	011010	011111
Input	010000	010001	010010	010011	010100	010101	010110	010111
Output	001000	011010	001010	011010	011010	011010	011010	011111
Input	011000	011001	011010	011011	011100	011101	011110	011111
Output	001010	011010	001010	011010	011010	011010	011010	011111
Input	100000	100001	100010	100011	100100	100101	100110	100111
Output	000000	010010	011000	011010	010010	010010	011010	011010
Input	101000	101001	101010	101011	101100	101101	101110	101111
Output	001010	011010	011010	011010	011010	011010	011010	011111
Input	110000	110001	110010	110011	110100	110101	110110	110111
Output	001000	011010	011010	011010	011010	111010	011010	111111
Input	111000	111001	111010	111011	111100	111101	111110	111111
Output	001010	011010	011010	011010	011010	111010	011010	111111

Nu moet ieder groepje een Alicia en een Benito kiezen. De groep kan in tweeën delen en de ene helft hoort bij Benito en de ander bij Alicia. Alicia moet een willekeurige input kiezen en daarmee de output bepalen en deze aan de groep Benito doorgeven. Benito gokt de pariteit van de input (heeft de input van Alicia een oneven of een even aantal enen). Het moet duidelijk zijn voor de leerlingen dat de groep Benito alleen een wilde gok kan doen en de gekozen input niet kan redeneren. Als de groep Benito heeft gegokt en de uitkomst heeft doorgegeven vertelt de groep Alicia haar input. Groep Benito wint als hij de juiste pariteit heeft gekozen en groep Alicia als de groep Benito verkeerd heeft gegokt. Benito kan controleren of Alicia haar gekozen input toch niet stiekem heeft veranderd door haar input in het circuit te stoppen en te kijken of daar dezelfde output uit komt in het circuit. Hiermee komt een einde aan het spel.

Benito kan vals spelen als hij met de gegeven output kan achterhalen wat de input was die deze output heeft gegenereerd. Dus het is voor Alicia van belang dat de functie van het circuit een eenrichtingsfunctie is, zoals besproken in Activiteit 14, zodat Benito niet kan valsspelen. Een eenrichtingsfunctie is een functie waarmee je makkelijk met een bepaalde input tot een output kan komen, maar met de output het heel moeilijk is om de input te achterhalen.

Alicia kan vals spelen als ze twee verschillende inputs met verschillende pariteit kan vinden die dezelfde output genereren. Dan kan ze, wat Benito ook gokt, altijd de input geven die ervoor zorgt dat Benito het mis heeft. Dus het is van belang voor Benito om er zeker van te zijn dat het circuit niet bij veel verschillende inputs dezelfde output genereert.

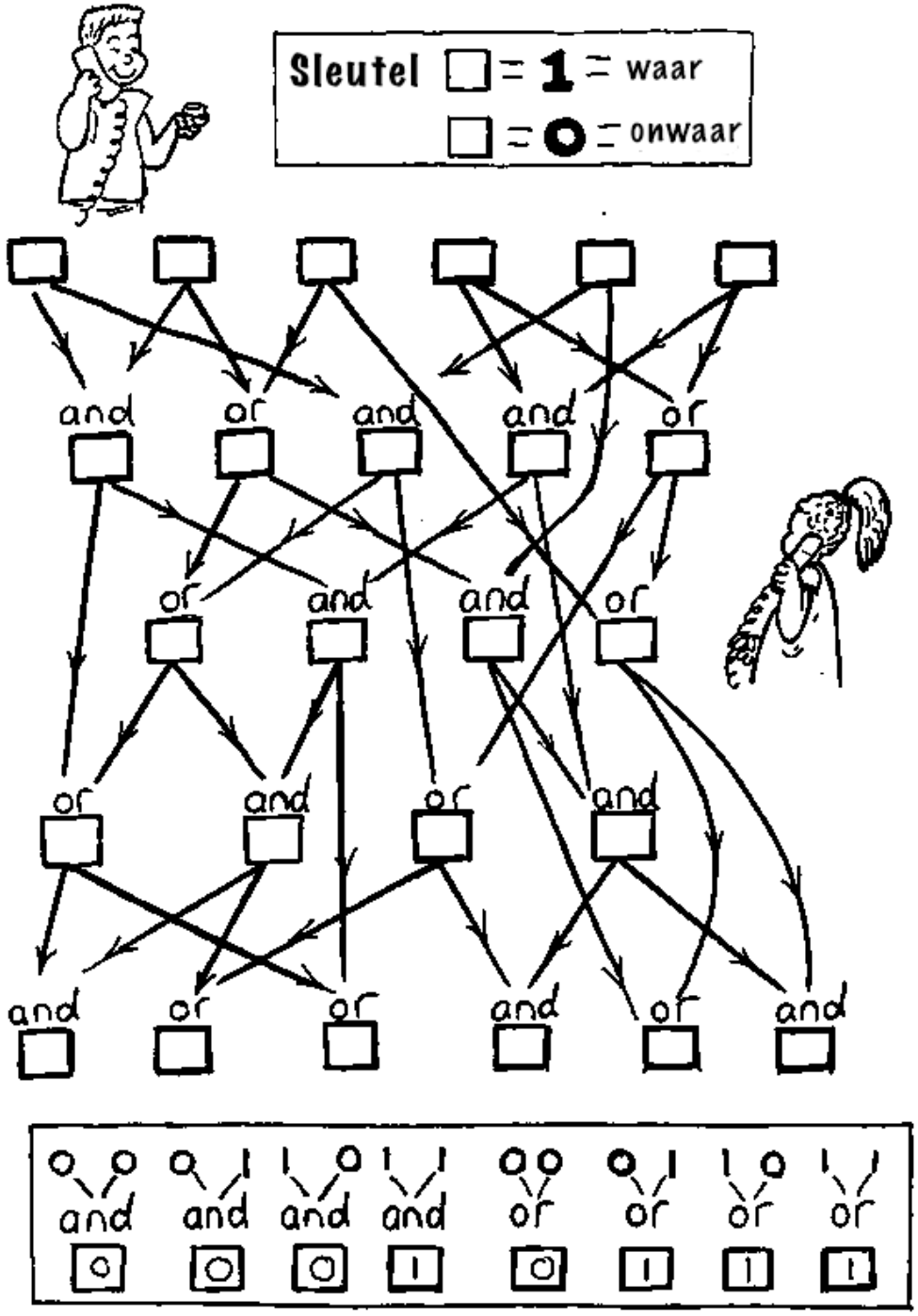
Kijk eens of de leerlingen een manier kunnen bedenken hoe Alicia of Benito vals kunnen spelen. Vanaf de eerste regel van de tabel zijn er meerdere inputs mogelijk om tot output 010010 te komen, bijvoorbeeld 000001, 000011, 000101 enzovoort komen tot deze output. Dus als Alicia Benito laat weten dat de output 010010 is kan ze zeggen dat ze input 000001 had gekozen als Benito gokt dat de pariteit even is en 000011 als hij gokt dat het oneven is.

Met dit circuit is het moeilijk voor Benito om vals te spelen. Maar als de output toevallig 011000 is dan moet de input 100010 zijn geweest. Er is geen andere mogelijkheid (je kan dit controleren door door de tabel te gaan). Dus als dit toevallig het nummer is waar Alicia mee komt, dan kan Benito een even pariteit kiezen en zeker weten dat hij het goed heeft. Een computer zal veel meer bits gebruiken en zullen er veel te veel mogelijkheden zijn om uit te proberen (iedere extra bit verdubbeld het aantal mogelijkheden).

Vraag nu de groepjes leerlingen om hun eigen circuits te ontwikkelen voor dit spel. Vraag een groepje of ze een circuit kunnen bedenken die het makkelijk maakt voor Alicia om vals te spelen en een ander groepje een om Benito makkelijk vals te laten spelen. Het aantal van zes inputs is arbitrair, de leerlingen mogen met andere aantallen in- en outputs netwerken maken.

Werkblad Activiteit: De Peruviaanse toss

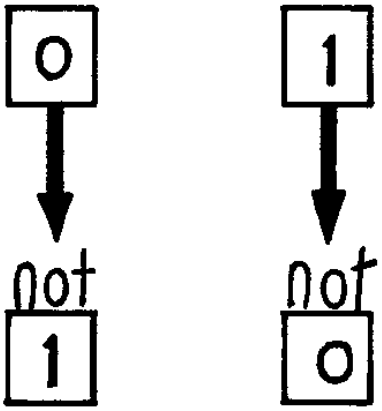
Kies een aantal inputs voor dit circuit en zoek uit wat de outputs zijn.



Variaties en uitbreidingen

In de praktijk gaat het dus om de samenwerking die nodig is om tot een circuit te komen dat voor Alicia en Benito acceptabel is. Dit is dan leuk voor een oefening in de klas, maar is hoogstwaarschijnlijk onbruikbaar in de praktijk om tot een gezamenlijke circuit te komen en zeker niet via de telefoon! Maar er is een makkelijk alternatief waarin Alicia en Benito onafhankelijk van elkaar een circuit maken en deze publiekelijk toegankelijk maken. Vervolgens stopt Alicia haar input in beide circuits en voegt de outputs samen door ze boven elkaar te zetten en met elkaar te vergelijken. Als het eerste getal van beide outputs gelijk zijn, schrijft ze een één op en als ze niet gelijk zijn een nul. En zo vergelijkt ze alle getallen van beide outputs om zo tot een nieuwe output te komen. Op deze manier kan geen van deelnemers vals spelen als de ander dat niet doet. Als een van de circuits een eenrichtingsfunctie is, is de combinatie van beide circuits dat ook.

De volgende twee variaties zijn niet direct gerelateerd aan cryptografische protocollen of het munt-toss probleem, maar gaan in op het feit dat circuits bestaan uit AND en OR poorten. Ze gaan dieper in op een aantal belangrijke ideeën in de basis van niet alleen computer circuit, maar meer logica zelf. Deze vorm van logica heet Booleaanse algebra, vernoemd naar de wiskundige George Boole (1815-1864).

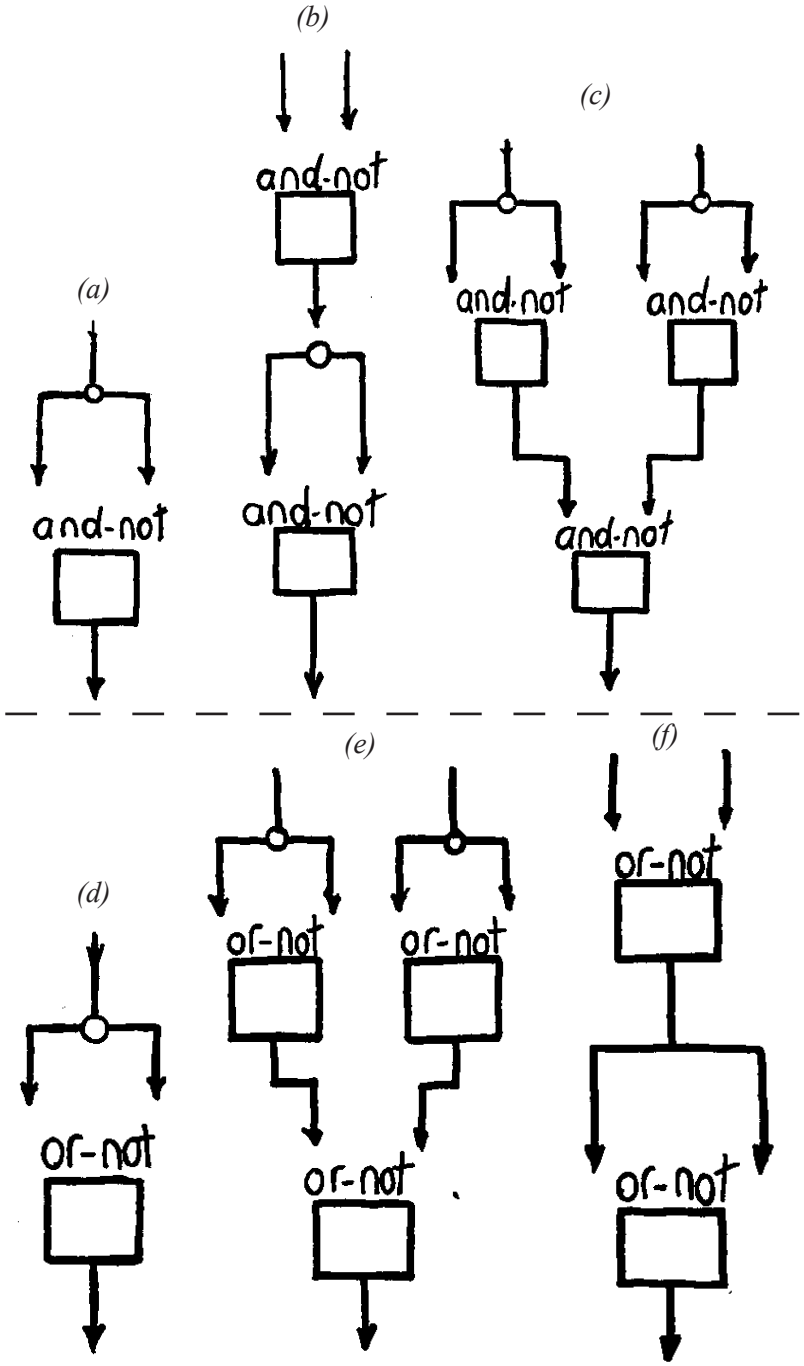


De leerlingen kunnen opgemerkt hebben dat een input met alleen maar nullen, 000000 leidt tot een output met alleen maar nullen en een input met alleen enen, 111111, leidt tot een output van alleen enen. (Er kunnen ook andere inputs die tot deze outputs leiden; bij het voorbeeld circuit leidt input 000010 tot allemaal nullen en 110111 tot allemaal enen). Dit is een gevolg van het feit dat de circuits bestaan uit AND en OR poorten. Door een NOT-poort toe te voegen, die een input neemt en het tegenovergestelde als output genereerd (bijvoorbeeld 0 wordt 1 en 1 wordt 0), kunnen de studenten circuits maken die niet deze eigenschap heeft.

Twee andere belangrijke soorten poorten die gebruikt kunnen worden zijn AND-NOT en OR-NOT (vaak afgekort als respectievelijk NAND en NOR), die precies zijn zoals AND en OR maar gevolgd door een NOT. Dus a AND-NOT b is hetzelfde als NOT(a AND b). Hiermee kunnen niet andere circuits gemaakt worden, omdat hetzelfde effect bereikt kan worden door een AND of OR poort gevolgd door een NOT. Hoe dan ook hebben ze wel de interessante eigenschap dat alle andere poort typen gemaakt kunnen worden uit AND-NOT poorten en ook uit OR-NOT poorten.

Nu er AND-NOT en OR-NOT poorten geïntroduceerd zijn kan je de leerlingen uitdagen om uit te zoeken of iedere poort gemaakt kan worden door andere poorten aan elkaar te

koppelen en vervolgens of ze dit kunnen maken van maar een type poort die aan elkaar gekoppeld wordt. De figuur hieronder laat zien hoe de drie basis poorten, de NOT, AND en OR gemaakt kunnen worden uit de AND-NOT poorten uit de bovenste rij (a, b en c) en de OR-NOT poorten in de onderste rij (d, e en f).



Waar gaat dit over?

De hoeveelheid geld en vertrouwelijke gegevens die over computernetwerken gaan is de laatste jaren enorm toegenomen en het is natuurlijk heel belangrijk om hierbij een veilige uitwisseling van elektronisch geld, getekende vertrouwelijke documenten en transacties te garanderen. Cryptografie gaat over het veilig en privé communiceren. Tientallen jaren geleden ontdekte computer wetenschappers dat geheimhouding het best gegarandeerd kan worden door technieken die er voor zorgen dat bepaalde informatie juist openbaar ofwel publiek is, al voelt dit natuurlijk wel tegennatuurlijk. Het resultaat van dit onderzoek heet "Publieke sleutel encryptie" (zie activiteit 18, Kid Krypto), wat nu wijdverbreid wordt gebruikt als belangrijkste veilige manier om gegevens uit te wisselen. Je hebt misschien wel eens gehoord van SSL (Secure Sockets Layer) of TLS (Transport Layer Security) in je webbrowser, deze systemen zijn gebaseerd op een publieke sleutel die je webbrowser in staat stelt een veilige verbinding te maken met een bank, ook al heeft iemand toegang tot alle data die wordt verzonden via deze verbinding, deze persoon kan niets met deze gegevens.

Cryptografie gaat niet alleen over dingen geheimhouden, maar ook over het plaatsen van controle over welke gegevens beschikbaar zijn en over vertrouwen tussen mensen die geografisch van elkaar gescheiden zijn. Formele regels of beter gezegd "protocollen" voor cryptografische transacties zijn ontworpen om op het oog onmogelijke dingen als niet te vervalsen digitale handtekeningen te verzenden en de mogelijkheid om anderen te vertellen dat je een geheim hebt (zoals een wachtwoord) zonder te verklappen wat deze is. Een munt opgooien via de telefoon is een simpeler en analoog probleem, dat op het eerste gezicht ook onmogelijk lijkt.

In de praktijk zullen Alicia en Benito niet zelf het circuit ontwerpen, maar een computerprogramma gebruiken die dit voor ze doet. Waarschijnlijk zullen ze beide niet erg geïnteresseerd zijn in hoe het programma van binnen werkt. Maar willen ze er allebei alleen maar van verzekerd zijn dat de ander niet de mogelijkheid heeft de uitkomst te beïnvloeden hoe goed hun computervaardigheden ook zijn en hoe vaak ze het ook proberen.
In principe zou ieder verschil van mening door een neutrale rechter opgelost moeten kunnen worden. De rechter zou beschikking moeten hebben tot het circuit, Alicia's originele binaire nummer, de output die ze naar Benito heeft verstuurd en de gok die Benito heeft teruggezonden. Als deze interactie voorbij is, is dit allemaal publieke informatie zodat beide deelnemers erover eens kunnen komen dat dit de juiste uitslag was. De rechter zal Alicia's originele nummer invoeren, de output checken en kijken of deze overeenkomt met de gegeven output en zo beslissen of de uitkomst eerlijk is.

Doordat het een duidelijke openbare procedure is, is de kans klein dat er een verschil van mening zal ontstaan. Vergelijk dat met de situatie waarbij Alicia een munt opgooit en

Benito door de telefoon kop of munt roept - geen rechter zal bij een verschil van mening deze zaak aannemen!
Een klein circuit zoals hier gebruikt is, is in de praktijk onvoldoende. Het is te makkelijk om een tabel te maken en deze te gebruiken om vals te spelen. Tweeëndertig bits als input gebruiken levert een veel betere bescherming op. Maar dit zal ook niet garanderen dat het heel moeilijk te kraken is - dat hangt weer af van het gebruikte circuit. Andere methoden zouden ook nog gebruikt kunnen worden, zoals de eenrichtings-functie in activiteit 14, Toeristenstad. De methoden die in de praktijk worden gebruikt, gebruiken vaak enorme getallen, wat dit tot een moeilijk probleem maakt (hoewel we in de volgende activiteit zullen leren dat deze niet NP-volledig is).

Het is makkelijk te checken of één getal een factor van een ander getal is, maar de factoren van hele grote getallen vinden kost heel veel tijd. Dit maakt het veel lastiger voor Alicia en Benito (en de rechter) om het met de hand uit te vinden.
Digitale handtekeningen zijn gebaseerd op hetzelfde idee. Door de output van de geheime input in het circuit publiek te maken is het makkelijk voor Alicia te bewijzen dat zij degene is die de output heeft gegeven - met een goede eenrichtings-functie kan niemand met een input komen die dezelfde output geeft. Niemand kan zich voordoen als Alicia! Om een werkelijke digitale handtekening te maken is een ingewikkelder protocol nodig om er zeker van te zijn dat Alicia een bepaalde boodschap kan ondertekenen en men moet kunnen checken of Alicia de ondertekenaar was, zelfs als zij beweert dat niet te zijn geweest. Maar het principe is hetzelfde.

Een andere toepassing is bij het spelen van poker over de telefoon in een omgeving waar geen scheidsrechter is om de kaarten te delen en de handen van beide spelers te controleren. Alles moet door de spelers zelf geregeld kunnen worden inclusief een toevlucht tot een rechter aan het eind van een spel in het geval van een geschil. Vergelijkbare situaties doen zich voor bij contractonderhandelingen. De partijen moeten hun kaarten geheim kunnen houden tijdens de onderhandeling. Maar ze moeten wel eerlijk blijven - ze mogen niet kunnen claimen dat ze een aas hebben zonder dat ze deze daadwerkelijk hebben! Dit kan gecontroleerd worden door te wachten tot aan het einde van het spel en dan elkaar toe te staan elkaar hand te controleren en de zetten die zijn gedaan. Een ander probleem is hoe je de kaarten moet uitdelen terwijl de handen van elke speler geheim blijven tot aan het eind van het spel. Verrassend genoeg is het mogelijk dit te bereiken door een cryptografisch protocol te gebruiken niet heel anders dan die bij de munt opgooi.
Cryptografische protocollen zijn heel erg belangrijk in elektronische transacties, of je nou de eigenaar van een bankpas moet verifiëren, een telefoon toestemming moet geven voor een verbinding of een zender van een email moet kunnen identificeren. De mogelijkheid om dit te doen is cruciaal voor het succes van elektronisch betalingsverkeer.

Verder lezen

Harel's boek Algorithmics bediscussieert digitale handtekeningen en bijbehorende cryptografische protocollen. Het laat ook zien hoe je een spelletje poker kunt spelen over de telefoon, een idee dat voor het eerst werd geopperd in 1981 in het hoofdstuk "Mental poker", in het boek The Mathematical Gardner, door D.A. Klarner. Cryptography and data security door Dorothy Denning is een excellente computer science tekst over cryptografie. Dewdney's Turing Omnibus bevat een sectie over Booleaanse logica dat de bouwstenen voor de circuits in deze activiteit behandelt.