



PROGRAMMERA MERA 2.0

LÄRARHANDLEDNING



FÖRFATTARE
KARIN NYGÅRDS

PROGRAMMERA MERA

Datorer finns överallt omkring oss. Allt som är digitalt är programmerat och mer och mer blir digitalt. Därför är det bra att förstå hur det fungerar. Så inleder programledaren Karin Nygårds varje avsnitt av Programmera mera 2.0, en serie som ska hjälpa unga att förstå tekniken som ligger bakom det som händer på skärmen.

ETT LUSTFYLLT PROGRAM OM PROGRAMMERING

Serien Programmera mera 2.0 vill inspirera elvaåringar och deras lärare att ta sig an programmering utifrån olika perspektiv. I åtta avsnitt på 15 minuter, varvas förklaringar, med frågor, utmaningar och en intervju med en vuxen "expert" från någon del av IT-sektorn. Tre elvaåringar möter programledaren Karin Nygårds i hennes studio, där hon låter dem testa att plocka ihop en dator, läsa riktig kod, lösa krypterade meddelanden och programmera robotar.

PROGRAMMERING – ÖVERALLT OMKRING OSS

Fokus i Programmera mera 2.0 är att sätta in programmering i ett sammanhang, snarare än att faktiskt lära ut programmeringsspråk. Serien ger inspiration till elever och pedagoger som vill lära sig grunderna i programmering utan att behöva förkunskaper eller teknisk utrustning. Det är viktigare att förstå hur det fungerar, än att faktiskt programmera.

TV-PROGRAM, LEKTIONSTIPS OCH HANDLEDNING

I handledningen finns förklaringar på hur de olika delarna i programmet hänger ihop, samt lektionstips till varje avsnitt. Förutom grundserien och handledningen finns åtta korta program, Programmera mera – Lektionstips, som beskriver tankarna bakom varje avsnitt och visar praktiskt hur en del av övningarna som finns beskrivna i handledningen fungerar i klassrummet. Om man är helt obekant med programmering, kan det vara en bra start att börja med att titta på Programmera mera – Lektionstips som du hittar på urskola.se.

Även om lgr11 bara har med programmering som centralt innehåll i matematik och teknik, kan det även inkluderas i ämnen som idrott och hälsa, svenska och samhällsorientering. I den här handledningen får du förslag som kan anpassas till olika ämnen.

TACK TILL

Linda Manilla Forskare inom datavetenskapens didaktik vid Linköpings universitet och Aalto universitet, Esbo, Finland. Johan Öbrink Digital Strateg och (enligt TechWorld) en av Sveriges bästa utvecklare.

TEMA

Varje avsnitt börjar med att programledaren presenterar avsnittets tema. Valet av teman grundar sig på intervjuer med ett hundratal elvaåringar, om vad de använder mest och vad som intresserar dem, när det kommer till den digitala delen av världen.

VISUALISERING

Som inledning till avsnittet visas en visualisering med anknytning till temat under rubriken "Fatta hur det funkar". Det är inslag, oftast visualiserade med hjälp av barn som själva utför till exempel de olika delarna i en webbsökning.

FRÅGA

En kort faktafråga som följer direkt efter visualiseringen, för att fånga upp vad inslaget handlade om.

UTMANINGAR

Deltagarna får två olika utmaningar att utföra på egen hand i studion. Utmaningarna har till viss del med avsnittets tema att göra, men vissa utmaningar återkommer också under serien som repetition och för att bidra till igenkänning och till vad tittaren faktiskt redan har lärt sig.

Tanken med utmaningarna är att deltagarna ska samarbeta och våga pröva. Det är alltså inte ett prov på vad de redan kan, utan handlar mer om att de ska visa hur de kan lösa de uppgifter de tilldelas. Tävlingsformen är vald för att öka spänningen för tittaren, men är inget som är nödvändigt att använda sig av i klassrummet. Likaså är bedömningen på poängstegen något nyckfull och ska inte jämföras mellan de olika avsnitten. Ett bra samarbete kan ge höga poäng, även om uppgiften inte klaras av helt korrekt. Tanken är att det ska kännas lekfullt och roligt. Den korta programtiden gör att deltagarna inte hinner försöka flera gånger, vilket annars är en viktig del av programmering. I klassrummet bör eleverna uppmuntras att prova om och om igen, tills de lyckas.

EXPERT

I varje avsnitt intervjuas en vuxen expert som jobbar med något kopplat till avsnittets tema. Vi får träffa åtta olika personer med skiftande bakgrund och med olika relation till programmering. Dessa möten ger en bred bild av vad det kan innebära att jobba med programmering och vill visa att det inte är just programmerare man måste bli för att behöva kunskap om programmering.

QUIZ

För att avrunda varje avsnitt får deltagarna svara på snabba faktafrågor som har med programmering och den digitala världen att göra. Oftast är det rena begreppsfrågor. De flesta frågorna finns det svar på i avsnittet, andra är mer generellt relaterade till programmering.

Tips! Använd gärna frågorna till spel eller frågesport i klassrummet till exempel genom att använda Kahoot, Padlet, Mentimeter eller andra responsiva program.

PROGRAMMERA MERA OCH STYRDOKUMENTEN

"Alla barn och elever behöver få förståelse för hur digitaliseringen påverkar världen och våra liv, hur programmering styr såväl det informationsflöde vi nås av som de verktyg vi använder, liksom att få kunskap om hur tekniken fungerar för att själv kunna tillämpa den."

Det inledande citatet är taget ur regeringens nationella strategi för skolans digitalisering som presenterades 19 oktober 2017. Det hade också kunnat vara texten som förklarar avsikten med Programmera mera. Syftet med programmet är att det ska sätta in programmering i en kontext, snarare än att lära ut förmågor och färdigheter. Det finns gott om bra resurser som lär ut programmering i dag. Med Programmera mera 2.0 vill vi visa på mekanismerna bakom skärmen och varför de är viktiga att förstå.

I läroplanen för grundskolan, presenterar Skolverket fyra aspekter av digital kompetens som ska genomsyra all undervisning i skolan.

- Att förstå digitaliseringens påverkan på samhället
- Att kunna använda och förstå digitala verktyg och medier
- Att ha ett kritiskt och ansvarsfullt förhållningssätt
- Att kunna lösa problem och omsätta idéer till handling

Innehållet i Programmera mera 2.0, tillsammans med lektionstipsen, syftar till att stärka samtliga fyra aspekter, genom att bidra med kunskap, förståelse och problematiserande frågeställningar.

CENTRALT INNEHÅLL SOM BEHANDLAS I PROGRAMMEN 1–3, 4–6, 7–9

MATEMATIK:

- Hur entydiga stegvisa instruktioner kan konstrueras, beskrivas och följas som grund för programmering. Symbolers användning vid stegvisa instruktioner.

- Hur algoritmer kan skapas och användas vid programmering. Programmering i visuella programmeringsmiljöer.
- Programmering i olika programmeringsmiljöer.

TEKNIK:

- Att styra föremål med programmering.
- Vad datorer används till och några av datorns grundläggande delar för inmatning, utmatning och lagring av information, till exempel tangenter, skärm och hårddisk. Några vanliga föremål som styrs av datorer.
- Att styra egna konstruktioner eller andra föremål med programmering.
- Några av datorns delar och deras funktioner, till exempel processor och arbetsminne.
- Hur datorer styrs av program och kan kopplas samman i nätverk.

SAMHÄLLSKUNSKAP:

- Hur man agerar ansvarsfullt vid användning av digitala och andra medier utifrån sociala, etiska och rättsliga aspekter.
- Digitaliseringens betydelse för individen, till exempel ökade möjligheter till kommunikation och elektronisk handel.
- Hur information i digitala medier kan styras av bakomliggande programmering.

TEMA	VISUALISERING	UTMANING 1	UTMANING 2	EXPERT	LEKTIONSTIPS	LEKTIONSTIPS
Datorer	Storlek på CPU	Binärkod med lampor	Montera dator – skapa låt	Cristobal Arneda, Project manager Bahnhof	Sortera input-output (teknik)	Imitera CPU-ALU-Skärm (teknik)
Mobilspel	Göra enkla spel	Logiska grindar –sortera godis	Robotbana	Jonna Ingolf, Narrative designer, King	Ändra 3 saker i ett spel (Programmering)	Programmera egna spel (Programmering + valfritt ämne)
Internet	Skicka bild över internet	Binärkod med lampor	Lösa krypterade ord	Anne-Marie Eklund Löwinder, Säkerhetschef, IIS	Spela upp hur internet fungerar (teknik)	Kryptoövningar (teknik/matematik)
Webbsök	Söka på webben	Ytliga/djupa webben	Robotbana	Olga Stern, Systemutvecklare, Olgatron	Homografer och olika resultat (SH)	Jämför olika sökmotorer och inställningar (SH)
Dataspel	Stora mängder kod	Läsa C++	Montera dator – skapa spel	Awed Haller Spe-animatör, Dice	Läsa C++ (programmering)	Skapa spel tillsammans i grupp (sv, bd, programmering)
Säkerhet	Attacker	Rangordna lösenord	Lösa krypterade ord	Siren Hofvander Chief Security Officer, Min Doktor	Beräkna säkerhet på lösenord (matematik, sannolikhetslära)	Tvåfaktorsinlogg –sortera olika slags faktorer (teknik)
Sociala medier	Instagrams algoritm	Vem ser vad jag delar?	Robotbana	Janne Elvelid Public Policy Manager Nordics, Facebook	Digitopia -så vill vi ha det (SH)	Analysera sociala medier (SH)
Robotar	Robothistoria	Logiska grindar –sortera godis	Robotdisco	Danica Kragic Jensfelt, Professor i robotik, KTH	Styra unplugged (Teknik)	Botmatte (matematik)

INNAN PROGRAMMET

Samtala med gruppen om vad en dator är. I dag finns det datorer i så många olika former att det inte alltid är självklart vad som är styrt av programmering och inte. Det övergripande syftet med avsnittet är att skapa förståelse för hur datorer fungerar. Att datorer är binära är en grundläggande begränsning för allt digitalt.

I avsnittet förekommer begreppen CPU, transistor, bit, ASCII, server, input, output, molnet, enkorts dator, RAM, grafikkort, minne samt CU (control unit) och ALU (arithmetic/Logic unit)

UNDER PROGRAMMET

VISUALISERING

Programledaren pratar om CPU, Central Processing Unit, alltså den del i datorn som tar hand om alla signaler in och ser till att rätt signaler går ut. I en CPU finns transistorer som processar ettor och nollor genom att de antingen släpper igenom ström (1) eller inte (0). I inslaget visas att antalet transistorer (hål i kannan) gör att dagens datorer kan processa många fler ettor och nollor, än de gamla, där det fanns färre transistorer.

FRÅGA

Vad gör en transistor?

Den reagerar på ett eller noll, alltså den släpper igenom ström (1), eller inte (0).

UTMANING I

Deltagarna ska tolka tre rader med tända/släckta lampor med hjälp av ASCII-tabellen. Varje tecken representeras av åtta lampor. Det finns både siffror och bokstäver att leta bland. Syftet med utmaningen är inte att lära sig ASCII-tabellen, utan att förstå att ett och noll representerar ström som är av eller på.

EXPERTEN

Avsnittets expert heter Cristobal Arnedo och jobbar i en serverhall. En server är en enhet som lagrar data. När vi lagrar data i "molnet" är det inte alls uppe i luften som namnet kan lura en att tro, utan i serverhallar som den här.

UTMANING 2 – BYGG EN DATOR

Deltagarna får en väska med komponenter som ska sättas ihop, utan beskrivning, för att sedan programmera en låt, när datorn är färdig.

Programmet som används heter Sonic Pi och är en kostnadsfri programvara till windows, macOS, RaspberryPi och Linux (ej till OSX) som finns att ladda ner på <http://sonic-pi.net/>

QUIZ

Vad heter det som brukar kallas datorns hjärna?

CPU (Central Process Unit)

Vad gör en transistor?

Läser av ett och noll, alltså släpper igenom ström, eller inte.

Vad är en "bit"?

En etta eller en nolla kallas för en bit. Åtta bitar är en byte.

Hur många ettor och nollor är ett tecken i binära ASCII-tabellen?

Åtta bitar

Är ett tangentbord output eller input?

Input

Deltagarna når Diamant, vilket är den allra högsta nivå som går att nå. Elit finns med för att visa att vi alltid kan bli bättre, hela livet.

Här följer tips på övningar att göra efter programmet. Välj de uppgifter som passar gruppen. Titta gärna på Lektionstips som hör till avsnittet, för att se hur övningarna kan göras.

OBS! Övningarna som eleverna gör är inte rättade och ska inte användas som facit för vad som är rätt eller fel.

DISKUSSION

- Experten Cristobal säger att programmering är som att bygga med lego, att det är bäst att börja ordentligt från grunden när man ska bygga ett program. På vilka fler sätt kan man likna programmering med lego? Kan vi likna programmering med annat vi gör? (skapande, problemlösande, metodiskt)
- Kan allting datoriseras/programmeras? Går det att programmera en stol? Ett mjölkpaket? VILL vi kunna programmera allt i vår omgivning? Vad är fördelarna och nackdelarna?
- Hur påverkas miljön av att mer och mer blir digitaliserat?

INPUT OCH OUTPUT

Kopieringsunderlag 1.1

Här gäller det att identifiera vilka olika sätt vi kan använda för att mata in data i en dator och på vilka sätt det kan presenteras för oss. I vissa enheter, som i en mobiltelefon, kan vi inte se skillnaden mellan in- och output, men funktionerna i den är ändå desamma.

Prata om olika sätt som vi kan få in data i en dator, och på vilka sätt vi kan få output. Ta gärna exempel som brandvarnare, tjuvlarm, hiss och annat som kanske inte är det vi vanligast tänker på när det kommer till datorer.

Tangentbord – CPU – skärm

Röksensor – CPU – brandlarm

Rörelsesensor – CPU – tjuvlarm som tjuter

Hissknapp – CPU – hissen åker till rätt våning

Gör tre högar, en för input, en för output och en för både input och output.

Använd bilderna i kopieringsunderlaget 1.1 eller samla ihop föremål att sortera.

CU-ALU-SKÄRM

Kopieringsunderlag: 1.2.1, 1.2.2, 1.2.3

Förutom att processa ettor och nollor måste CPU också hålla reda på beräkningar. Den delen av CPU kallas ALU, (Arithmetic/logic Unit). Den andra delen heter CU (Control Unit). Nu ska vi se hur dessa samarbetar för att kunna meddela skärmen vad den ska visa. När vi rör med musen över en skärm så beräknas nya x,y-koordinater. I övningen simulerar eleverna processen som sker (eller ett par delar av den).

ELEVERNA JOBBAR TRE OCH TRE.

En elev är CU, en elev är ALU, en elev är skärm. Om någon grupp bara är två, kan CU och ALU vara samma person.

Förklara de olika delarnas funktioner i datorn.

CU säger till ALU vad den har fått för indata.

ALU skriver ner värden på x och y och beräknar nytt värde, säger till CU.

CU meddelar skärmen vad den ska rita ut.

Skärmen markerar i sitt papper.

CU säger nästa värde till ALU.

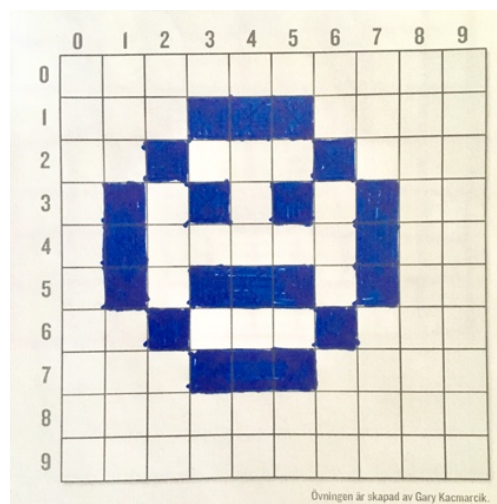
ALU beräknar nytt värde och meddelar de nya koordinaterna till CU.

CU meddelar skärmen osv.

Kontrollera gärna med jämna mellanrum att ALU har koll på uträkningarna. Om det blir ett fel så blir allt fel.

FACIT:

5 3	5 7	2 6
6 6	3 7	2 2
3 5	1 5	6 2
1 3	1 4	3 3
5 1	4 7	3 1
7 3	7 5	4 1
4 7	5 5	4 5



Övningen är skapad av Gary Kacmarcik.

AVSNITT 2 TEMA MOBILSPEL

INNAN PROGRAMMET

Övergripande syfte med avsnittet är att hitta en enkel väg in i programmering. Mobilspel är oftast mycket enklare i sin struktur, än stora datorspel, vilket gör att de kan vara en bra ingång till att prova på programmering med hjälp av övningarna som anknyter till avsnittet.

I avsnittet förekommer begreppen: loop, algoritm, logiska grindar, casual games, narrative designer

UNDER PROGRAMMET

VISUALISERING

Inslaget visar hur vi kan uppfatta roboten som dansar samma slinga (loop) om och om igen som helt olika "spel". Tanken är att inspirera tittarna till att våga använda ett och samma grundspel för att variera sig. Alltså bygga nya historier, istället för att komma på nya spel.

FRÅGA

Vad är en algoritm?

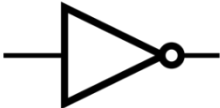
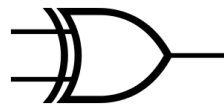
Det är som ett slags recept över hur man ska lösa ett programmeringsproblem. I visualiseringen såg vi prov på samma algoritm som återanvändes på olika sätt.

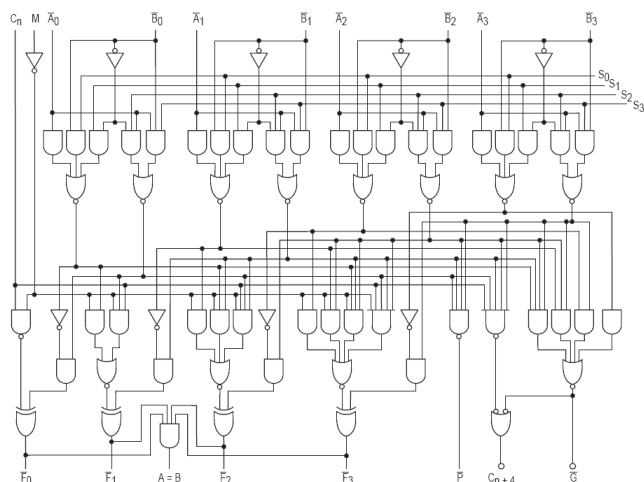
UTMANING I

Deltagarna ska sortera godis efter olika preferenser, symboliserat av logiska grindar.

Det är inte så tydligt i uppgiften, vad logiska grindar är. Det är symboler för hur ett eller noll ska tolkas i en dator, alltså, om en transistor ska släppa igenom ström eller inte. Det är rätt avancerat och svårt att förstå, så lägg inte någon längre tid på att förklara. Syftet med att ha med dem, är för att visa upp ett sätt att rita upp system för hur datorer ska tolka data. Det går självklart bra att fördjupa sig i detta mer för den som vill, till exempel inom fysiken eller tekniken.

DET HÄR ÄR VAD DELTAGARNA HAR ATT FÖLJA I SORTERINGEN:

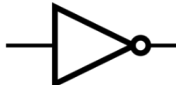
			
OCH /AND	EJ / NOT	ELLER / OR	ANTINGEN ELLER/ XOR
Måste vara två ettar eller två nollor för att bli sant.	Ett blir noll, noll blir ett.	Kan vara ett eller noll, spelar ingen roll.	Kan bara vara antingen ett eller noll. Inte både delarna.



Det här är en beskrivning över hur ALU, alltså den del av CPU som hanterar beräkningar, fungerar. Eleverna behöver verkligen inte förstå detta, men det kan vara bra att som pedagog ha ett hum om hur det ser ut när det används.

Källa:

https://en.wikipedia.org/wiki/Arithmetic_logic_unit

AMALIA	BÖRJE	CEASAR	DENA	EDDAN	FRANS
					
OCH /AND	EJ / NOT	ELLER / OR	ANTINGEN ELLER/ XOR	ELLER / OR	ANTINGEN ELLER/ XOR
Både skum och kvadrat.	Inte lakrits.	Gelé eller cirkel spelar ingen roll.	Antingen oval eller lakrits. Inte både delarna.	Kvadrat eller rektangel, spelar ingen roll.	Antingen skum eller gelé. Inte både delarna.

EXPERTEN

Avsnittets expert heter Yonna Ingolf och jobbar som narrative designer på ett företag som utvecklar mobilspel, eller casual games som Yonna säger i inslaget. En narrative designer skapar berättelsen i spelet och ser till att allt hänger ihop, med miljö och karaktärer. Utmaningen just i mobilspel är att man har en sån liten yta på sig att berätta en historia på.

UTMANING 2

Robotprogrammering. Deltagarna ska programmera en robot att ta sig igenom en rätt besvärlig bana, utan att hamna i vattenhinder eller sandlåda. Roboten programmeras via en datorplatta, som sköts av en deltagare. Därför är det extra viktigt att gruppen kan samarbeta bra. Det finns två olika svängar, 45 och 90 grader. Det gäller att välja rätt. Deltagarna misslyckas och roboten åker ner i en sandbunker nästan direkt.

Fråga eleverna vad de tror att deltagarna gjorde för fel.

Några möjliga orsaker:

- Det är inte så enkelt att komma ihåg allt och hålla reda på i huvudet vad man har programmerat. Oftast delar man upp problem i mindre bitar, istället för att försöka lösa allt på en gång.
- Mätte de tillräckligt noga? En centimeters fel i varje steg ger rätt många centimeter fel efter 20 steg.
- Har de kollat koden ordentligt innan de kör programmet?

QUIZ

Vad heter det som kallas för recept när man programmerar?

Algoritm

För att skriva kortare kod kan man använda en slinga?

Vad betyder det?

Loop, upprepning

Vad är logiska grindar?

Hur ettor och nollor tolkas.

Vad betyder den här?  NOT

Vad betyder den här?  OCH

EFTER PROGRAMMET

Här följer tips på övningar att göra efter programmet. Välj de uppgifter som passar gruppen. Titta gärna på Lektionstips som hör till avsnittet, för att se hur övningarna kan göras.

OBS! Övningarna som eleverna gör är inte rättade och ska inte användas som facit för vad som är rätt eller fel.

DISKUSSION

- Vad är det som gör ett spel bra? Fundera på vilka spel som är eller har varit mest populära. Vad har de gemensamt?
- Yonna berättar att hennes svåraste utmaning i jobbet är att kunna berätta mycket på ett litet utrymme. På vilka sätt kan man förmedla känslor i ett mobilspel?
- Spel används oftast som ren underhållning och tidsfördriv. Går det att lära sig något av att spela spel? Kan vi ha nytta av mobilspel? tips: cancerforskare använder sig av spelare för att snabbare kunna studera hur cancerceller fungerar <http://www.cancerresearchuk.org/about-us/cancer-news/press-release/2014-02-04-cancer-research-uk-launches-spaceship-smartphone-game-to-see-cancer-cures>

ÄNDRA ETT BEFINTLIGT SPEL

En väg in i programmering kan vara att börja med att ändra i befintlig kod. Här får eleverna i uppgift att försöka lära sig hur Scratch fungerar, genom att ändra tre olika saker i koden. Eleverna i lektionstipset fick ingen introduktion innan om hur Scratch fungerar, utan fick själva utforska på vilka sätt de kunde ändra. Några elever kände till hur man gjorde och kunde hjälpa de andra. Det går att göra samma övning i vilken programmiljö som helst, där det redan finns befintliga projekt att utgå från.

I scratch-studion för Programmera mera 2.0 finner du de projekt som används i webbinslaget:

<https://scratch.mit.edu/studios/4262899/>

REDOVISA PROJEKT I FRÅGESPEL

I samma scratchstudio som ovan, finns en mall för ett enkelt frågespel. Använd det för att låta eleverna redovisa olika arbetsområden under en SO- eller svensklektion utifrån olika böcker. Scratch är ett kul sätt att redovisa på, men innan programmeringen sitter, tar det mycket tid om eleverna ska börja helt från början. Då är det bra att börja med en gemensam mall.

GÖR DITT EGET FLAPPY BIRD

Det här är ett alternativ om ni inte har tillgång till egna datorer. På <https://studio.code.org/flappy/1> går det att göra sitt eget Flappy Bird-spel. Det brukar vara nyttigt att få prova sig fram till vad som gör ett spel bra. Code.org fungerar på alla plattformar, inklusive mobiltelefoner, även om det blir lite plottrigt just där.

INNAN PROGRAMMET

Vad är egentligen internet och hur fungerar det? Ta er gärna tid till att låta eleverna själva beskriva det i text och bild. Det är intressant!

I undersökningen Svenskarna och Internet (IIS 2017) uppger 89 % av alla 10 och 11-åringar att de använder internet varje dag. Vad använder de det till? Så här säger samma undersökning:

96 % tittar på film, 97 % spelar spel, 80% använder läroappar. 93 % söker information 88 % gör skolarbete, 93% skickar bilder och meddelanden, 81 % besöker sociala nätverk. Hur stämmer det med elevernas användning?

I avsnittet förekommer begreppen ASCII, DNS, domännamnssystem, IP-adress, IP-paket, IP-protokoll, port, protokoll, router

UNDER PROGRAMMET

VISUALISERING

En bild som skickas över internet består av miljoner ettor och nollor. Dessa delas in i mindre enheter som kallas paket. Varje paket förses med information om varifrån det kommer, vart det ska, vad det är för slags information (film, text, bild) och hur det ska läsas in. Paketen skickas via routrar till rätt adress. Det här är själva grundfunktionen i att skicka information över internet. Det heter att det följer Internet Protokollet. Paketen kallas IP-paket och adresserna IP-adresser. Porten, som informationen kommer till, är inte en fysisk port som går att se inne i datorn. Det är ett sätt att läsa in informationen.

FRÅGA

Vad är ett IP-paket?

Allt som skickas över internet delas in i mindre delar som kallas paket, så att det blir lättare att skicka.

UTMANING 1

Deltagarna ska skriva tre tecken, 4NZ, i binärkod, genom att tända och släcka lampor. Till sin hjälp har de ASCII-tabellen. Varje tecken representeras av åtta lampor som är tända för ett och släckta för noll.

I det här avsnittet har några rader i tabellen maskrats för att tvinga deltagarna till att tänka efter vilket mönster som den binära talföljden bildar. För den som vill kan det här vara en bra ingång till att prata om olika talbaser inom matematiken.

EXPERTEN

Anne-Marie Eklund Löwinder är säkerhetschef på Internetstiftelsen i Sverige. Det är en stiftelse som bland annat hanterar domännamnen .se och .nu. Bara en sida kan ju ha urskola.se, annars skulle det bli rörigt.

Tips! Kika gärna på iis.se om du som pedagog inte känner till Internetstiftelsen. Där finns väldigt mycket bra information och kurser för pedagoger och elever.

Anne-Marie är också krypto-officer. Det innebär att hon har en nyckel som håller internets domännamn säkra. När vi skriver in urskola.se så vill vi komma till just den sidan och ingen annan. För att säkerställa att det är rätt sida som är kopplad till rätt adress, finns ett säkerhetssystem som skyddas av krypteringsnycklar. Nyckeln som Anne-Marie har är en helt vanlig liten skåpsnyckel som behövs för att komma åt kodkort. De behövs för att kunna starta maskinen som gör krypteringsnycklar – alltså långa koder av siffror och andra tecken.

För att det ska gå att göra nya krypteringsnycklar behövs det sju personer från olika delar av världen. På så sätt kan inte en enda person, en grupp personer eller ett enskilt land, ta kontroll över internet.

Vill du veta mer? Nyckeln till internet är en kortfilm som finns på Internetmuseum. <https://www.youtube.com/watch?v=gQ07Ti4uSnQ>

UTMANING 2

Eftersom allt vi skickar över internet är helt öppet för vem som helst att läsa, för den som vill och kan, är det viktigt att kunna vara hemlig om det behövs. Då kan man skicka text eller filer som är krypterade. Ett krypterat mail går inte att läsa om man inte har rätt krypteringsnyckel. Det finns väldigt många olika sätt att kryptera på. Här har vi använt ett klassiskt ceasar-krypto för att göra orden svårlästa. Deltagarna har tre olika nycklar att välja mellan för att kunna läsa de fyra krypterade orden.

QUIZ

Vad kallas paketen som datan delas upp i när den skickas över internet?

IP-paket

Vad betyder det att något är krypterat?

Att det inte går att läsa om man inte har rätt nyckel.

Ge ett exempel på något som brukar krypteras på en dator?

Lösenord eller bankkod t.ex.

Vad är klartext?

Det som är skrivet så att man kan läsa.

Vad kallas tabellen som ni använde för att skriva med ettor och nollor?

ASCII-tabellen.

Hur många ettor och nollor representerar ett tecken enligt ASCII?

8 (Deltagarna svarar 7. Från början var det faktiskt bara 7 ettor och nollor i ASCII-tabellen men en nolla lades till, för att det skulle bli en hel byte, dvs åtta bitar. På sätt och vis har deltagarna rätt, men det vet de ju inte om, därför får de inte rätt för svaret).

Vad brukar man kalla det när datorn ska välja mellan olika alternativ?

Villkor

EFTER PROGRAMMET

Här följer tips på övningar att göra efter programmet. Välj de uppgifter som passar gruppen. Titta gärna på Lektionstips som hör till avsnittet, för att se hur övningarna kan göras. OBS! Övningarna som eleverna gör i filmerna är inte rättade och ska inte användas som facit för vad som är rätt och fel.

DISKUSSION

- Vilka är fördelarna och nackdelarna med internet?
- På vilket sätt har vårt samhälle påverkats av att vi har tillgång till internet?
- Vad skulle hända om internet slutade fungera i morgon? Hur skulle ditt liv förändras? Hur skulle det påverka vårt samhälle? Vilka funktioner skulle vi sakna mest? På individnivå och på samhällsnivå? Hur skulle skolan påverkas om internet försvann?

SKICKA MEDDELANDEN ÖVER INTERNET

Kopieringsunderlag 3.1.1 och 3.1.2

Den här övningen kan göras på flera olika sätt, beroende på gruppens förmåga och tillgång till tid och rum. Syftet är att visa att allt som skickas över internet delas upp i mindre bitar, tar olika vägar och har adresslappar på sig.

Det som ska skickas kan vara en bild eller en text klippta i flera bitar. Välj hur många meddelanden som ska skickas och hur många kuvert de ska delas upp i. Tre kuvert till varje meddelande brukar fungera bra, men det går att variera så att det passar gruppen.

Lägg en bit i varje kuvert och skriv URL-adressen och vilken plats i meddelandet biten har. Till exempel:

urskola.se

google.com

Nummer 1

övre vänstra hörnet

EXEMPEL PÅ GENOMFÖRANDE

Placera ut elever med lappar med IP-adresser (kopieringsunderlag 3.1.2) i ena änden av rummet.

- Lägg alla kuvert som ska skickas i en hög, i andra änden av rummet.
- Välj om en elev ska vara DNS, som håller i kopieringsunderlag 3.2.1, eller om DNS-tabellen ska finnas projicerad på duk.
- På given signal springer eleverna och hämtar varsitt kuvert, kollar i DNS-tabellen vilken IP-adress det ska till och springer till rätt destination.
- När alla kuvert har kommit fram, ska de öppnas och pusslas ihop i rätt ordning.

I programmet Programmera mera – Lektionstips är två elever "hinder" på vägen. De symboliserar dålig mottagning, trasiga kablar eller överbelastning på grund av mycket trafik. Om routern som har skickat meddelandet inte får en bekräftelse, skickas paketet på nytt. Eftersom paketen skickas igen, och alltså bara fördröjs, så får hindren bara stoppa paketen i tre sekunder innan de får springa vidare. Det här momentet är valfritt att ta med, men det brukar uppskattas och ge lite extra spänning i uppgiften.

KRYPTERING

Kopieringsunderlag 3.2

- Gör ett krypto med alfabetet + 3, alltså A blir D, B blir E osv.
- Gör ett krypto med alfabetet baklänges.
- Gör ett krypto med hjälp av en hemlig fras. Till exempel: "programmera mera".
- Skriv in varje bokstav från frasen, en gång. Resten av kryptot fylls på med övriga alfabetet. Försök hitta så bra fras som möjligt så att så många bokstäver blir omkastade.
- Hitta på ett eget sätt att konstruera ett krypto på.
- Hitta på ett sätt att förmedla nyckeln utan att ge facit. Till exempel: A+3 för kryptot som bygger på alfabetet + 3 bokstäver.
- Skriv meddelande och byt med en kompis.
- Lös varandras krypterade meddelanden.

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	Å	Ä	Ö
P	R	O	G	R	A	M	B	C	D	F	H	I	J	K	L	N	Q	S	T	U	V	W	X	Y	Z	Å	Ä	Ö

INNAN PROGRAMMET

Övergripande syfte med avsnittet är att förklara hur programmering ligger bakom resultaten vi får när vi söker på webben. Det kan vara bra att inleda med att prata om skillnaden mellan internet och webben. Webben heter egentligen World Wide Web och skapades av Tim Berners Lee 1990, för att göra det lättare att använda internet.

Internet är vägarna mellan olika IP-adresser. Webben kan liknas vid hus som vi besöker. För att använda webben måste vi använda internet, men vi kan använda internet utan att använda webben. Det gör vi till exempel när vi skickar mail från dator till dator, men det finns förstås webbaserad mail också.

Alla webbsidor har unika adresser, URL (Uniform Resource Locator). Alla adresser finns i ett gigantiskt register, som på ett bibliotek. För att de ska kunna hittas måste varje sida indexeras. När man väljer att inte göra ett youtubeklipp sökbart, betyder det att det inte är indexerat. Nästan allt innehåll blir indexerat om vi inte väljer att hålla det hemligt. Ungefär 4,5 miljarder sidor är sökbara, men det finns alltså mer än dubbelt så många sidor till, som är dolda.

I avsnittet förekommer begreppen: djupa och ytliga webben, indexering, sökmotor, URL, webbcrawlers, webben.

UNDER PROGRAMMET

VISUALISERING

Inslaget liknar webben med ett bibliotek, där böckerna är webbsidor. Bibliotekarien får vara sökmotorn, som får in sökordet och skickar ut sökrobotarna att leta igenom biblioteket i jakt på relevanta sidor.

FRÅGA

Vad kallas sökrobotarna som letar igenom webben?
Webcrawlers. På svenska kallas de också för sökspindlar, eller bara spindlar. Det är inte några riktiga robotar, utan program som söker igenom webben.

UTMANING 1

Deltagarna ska sortera olika exempel på webbsidor, utifrån om de är sökbara eller inte. Sidor som man kan hitta i en webbsökning tillhör den ytliga webben, medan de som inte går att komma till utan inloggning, eller att man vet den exakta adressen, kallas för den djupa webben.

I övningen illustreras hela webben av ett isberg, eftersom det bara är en liten del av allt innehåll på webben som är synligt. Det mesta tillhör den djupa webben. Om man använder husliknelsen igen, så är det som att bankens hemsida är huset vi alla kan komma till, men bakom husets fasad finns tusentals sidor som vi inte når eftersom det är människors bankkonton. Det är webbsidor, men de går inte att googla sig till.

EXPERTEN

Avsnittets expert heter Olga Stern och är programmerare. Hon har bland annat gjort ett program som kan räkna antalet kvinnor och män i nyheter, för att kunna visa hur pass jämställda nyheterna är. Olga jämför programmering med stickning. Det är en bra liknelse på flera sätt: man skapar något själv, man måste lösa problem och komma på hur man ska få till exakt det man vill och man följer instruktioner som faktiskt ser rätt mycket ut som kod. Stickmönster innehåller funktioner och loopar, precis som programmeringskod och det gäller att göra exakt rätt, annars blir det hål i stickningen. Det var också inom textilindustrin som de första hålkorten användes.

UTMANING 2

Deltagarna ska programmera en robot att ta sig runt på en bana, över mållinjen, utan att hamna i sandbunkrar eller vattenhål. Koden skrivs in av en av deltagarna, på en datorplatta. De andra två deltagarna mäter och beräknar hur roboten ska gå, och säger vilken kod som ska skrivas in. Här gäller det att samarbeta bra. Banan är klurig och bjuder på många svårigheter. Bland annat måste deltagarna hålla reda på vilken slags svängar som den ska göra, 45 eller 90-grader?

QUIZ

Vad heter de små botarna som är programmerade att hitta sökord på webben?

Webcrawler

Vilka två olika delar av webben finns det?

Den ytliga och den djupa.

Vilken är störst?

Den djupa

Nämn något som finns i den djupa webben.

Mail och lösenordsskyddade sidor

Nämn något som går att hitta i den ytliga webben.

Alla öppna hemsidor

EFTER PROGRAMMET

Här följer tips på övningar att göra efter programmet. Välj de uppgifter som passar gruppen. Titta gärna på Lektionstips som hör till avsnittet, för att se hur övningarna kan göras. OBS! Övningarna som eleverna gör i programmet är inte rättade och ska inte användas som facit för vad som är rätt eller fel.

DISKUSSION

- På vilket sätt påverkas sökresultat av programmering?
- Vad är det som är bra med att vi får individanpassade sökresultat?
- Vad är nackdelarna med anpassningarna?
- Många av de algoritmer som styr våra resultat är baserade på fördomar. Hur kan det påverka oss i framtiden? Lästips till: www.di.se/nyheter/sara-ohr-vall-algoritmer-som-diskriminerar/

DEM FÅR VILKET RESULTAT? HOMOGRAFER

Kopieringsunderlag 4.1

Ett sätt att undersöka hur olika sökhistorik påverkar resultaten, är att försöka få fram olika resultat för olika intressen. Homonymer är ord som stavas lika, men uttalas och betyder olika saker. Beroende på vem det är som skriver in ordet i sökmotorn, kommer resultatet att se olika ut.

Övningen går ut på att hitta två olika slags innehåll som kan intressera två olika fiktiva sökare. Det är lättast att ha sökningen inställd på bilder. Eleverna måste klura ut vilket slags innehåll de olika betydelseerna av orden kan tänkas innebära.

JÄMFÖRA SÖKMOTORER

Kopieringsunderlag 4.2

Uppgiften går ut på att skriva in samma sökord i olika sökmotorer. Det är lättast att välja bildresultat, eftersom det blir tydligast. Speciellt på olika språk.

Om man väljer politiker, som Trump, Putin eller liknande, är det lättare att föra en diskussion om ensidighet och hur sökresultaten speglar våra värderingar. Vilka bilder lyfts fram i de olika sökmotorerna? Här är det extra intressant att jämföra sökmotorer från andra länder.

bing.com	microsofts söktjänst
duckduckgo.com	söktjänst som inte sparar någon webbhistorik och alltså är anonym
baidu.com	kinesisk söktjänst
yandex.com	ryska söktjänst
thinga.com	amerikansk söktjänst för barn
sökis.se	svensk söktjänst för barn

Kom ihåg! Om eleverna inte har sina egna datorer kan det hända att det inte ger tydliga utslag i skillnader mellan vanlig sökning och inkognitoläge.

INNAN PROGRAMMET

Dataspel kräver knappast någon närmare introduktion. De flesta elever känner till det som kallas AAA-spel (trippel A games), även om de kanske inte vet att de kallas just det. I grafik och innehåll är spelen mer filmlika än mobilspel oftast är. Mobilspel är till för att roa i korta stunder och ofta. AAA-spel är gjorda för att man ska gå in i en annan värld och ägna flera timmar åt att spela.

Övergripande syfte med avsnittet är att visa att det ligger mycket arbete och utveckling bakom de spel och tjänster som vi använder. För att kunna ha ett kritiskt och ansvarsfullt förhållningssätt till den digitala världen krävs förståelse för att det är människor som har skapat den.

I avsnittet förekommer begreppen: animatör, sprite, loop, koordinater

UNDER PROGRAMMET

VISUALISERING

Inslaget visar hur mängden kod som ligger bakom dataspel har utvecklats sedan Pong, ett av de första spelen. I ett AAA-spel finns det kod som styr animationer, funktioner, ljud, effekter, spelfunktioner och navigation.

FRÅGA

Hur många ettor och nollor består ett spel av (ungefär)?
400 miljarder

Hur mycket är 400 miljarder?

Jämför: hur lång tid tar det att räkna till en miljard? Om det tar 2 sekunder för varje tal, tar det 2 miljarder sekunder, som är ungefär 63 år.

UTMANING I

Deltagarna ska läsa kod till ett spel som en spelutvecklare har skrivit i språket C++. Det är bara två små snuttar av all kod som krävs för att göra ett helt spel. Deltagarna ska leta efter olika saker i koden och markera med en magnet, när de hittar det som efterfrågas.

Det första de ska leta efter är vilken kod som programmerar fienden i spelet. Figurer och föremål som ska programmeras kallas ofta sprite i programmeringsspråk. När man programmerar använder man sig av koordinater som talar om var spritarna ska vara.

EXPERTEN

Avsnittets expert heter Awed Haller och är spelanimatör på ett stort spelföretag. Det är han som gör att spelen är så snygga och att figurerna ser så levande ut. För att utveckla ett spel krävs det att ungefär 150 personer arbetar under två år. Inte undra på att det är svårt att göra egna spel som är lika snygga!

UTMANING 2

Deltagarna får en väska med komponenter som ska sättas ihop, utan beskrivning, för att sedan programmera en egen version av spelet Pong.

Pong är ett av de enklaste spelen att programmera och det går att göra i de flesta programmeringsmiljöer - som Scratch till exempel. I programmet använder vi <http://www.pong.kano.me/> som guidar steg för steg hur man ändrar spelet.

QUIZ

Ni fick läsa riktig programmeringskod idag, vad hette språket?

C++

Vad heter det som anger position på skärmen som har X&Y-värden?

Koordinater

Vad kallas figurer i programmeringsmiljöer?

Sprite

EFTER PROGRAMMET

Här följer tips på övningar att göra efter programmet. Välj de uppgifter som passar gruppen. Titta gärna på Lektionstips som hör till avsnittet, för att se hur övningarna kan göras.

OBS! Övningarna som eleverna gör är inte rättade och ska inte användas som facit för vad som är rätt eller fel.

DISKUSSION

- Dataspel är en stor industri idag. Det finns flera svenska företag som gör spel som spelas över hela världen. Kan spel påverka samhället vi lever i?
- Spelar det någon roll vem som gör spelen?
- Om undervisningen i skolan innehöll mer spel – vad skulle vara bra, mindre bra eller kanske dåligt med det?
- I dag finns det gymnasieskolor som har speciella inriktningar för e-sportspelare, precis som för fotboll, ridning eller skidåkning. Ändå räknas inte e-sport som en sport på riktigt. Vad är sport egentligen? Varför finns det sportinriktningar på gymnasier? Läs gärna mer om vad Riksidrottsförbundet säger om e-sport.

LÄS KODEN

Arbetsblad 5.1 (2 sidor)

Uppgiften går ut på att försöka förstå lite av vad det står i kodsuttarna från spelet Tiny Shooter av Magnus Runesson. Programspråket heter C++ och är ett av de vanligaste programspråken som används i dag. Gör övningen som en enskild uppgift, i par eller i helklass.

Hur skiljer sig programmeringskod från vanlig text?

- Olika färger på olika delar av texten (jämför med om vi skulle skriva alla verb i en färg, alla substantiv i en annan)
- Annorlunda indrag

- Annan användning av stora/små bokstäver (versaler och gemena signalerar olika saker och ger olika funktioner i programmet)
- Annan användning av skiljetecken (; för avslut istället för punkt)
- { } används inte så ofta i vanlig text

VAD KAN VI KÄNNA IGEN FRÅN VANLIG TEXT?

- Flera ord på engelska
- Det finns en grammatik

LÄS IGENOM KODEN OCH ÖVERSÄTT DE ORD NI KÄNNER IGEN.

Utifrån de engelska orden, vad tror ni att koden styr i spelet? (Koden styr fienden, vad som händer om fienden blir träffad, var fienden ska vara osv)

LETA EFTER FÖLJANDE

Figurer eller föremål som kan programmeras kallas ofta sprite. Hur många gånger hittar ni ordet sprite i koden? (4)

För att styra var olika sprites ska finnas på skärmen använder man sig av koordinatsystem med x och y-positioner. Kan ni hitta var i koden det står om positioner?

Programmeringskod innehåller ofta villkor som talar om vad programmet ska göra OM något händer. I de flesta programspråk skrivs villkor med if. Hur många if kan ni hitta i koden? (13)

Tecknen // betyder att datorn ska ignorera texten som följer. Det är alltså text skriven till människor och inte till datorn som ska köra programmet. Programmerare använder det för att skriva anteckningar som förklarar koden. Leta efter en sådan anteckning.

SAMARBETA KRING ETT SPEL

Kopieringsunderlag 5.2

När ett AAA-spel (triple A-game) skapas är många inblandade. Det är olika personer som gör olika delar av spelet. Alla gör det som de är bäst på. Samarbetet gör spelet bättre. I skolan behöver alla få träna på de olika delarna, men det är viktigt att man delar upp arbetet ordentligt och gör en plan för hur samarbetet ska gå till.

Det finns olika sätt att styra samarbetet på. Kopieringsunderlag 5.2 ger ett exempel på hur arbetet kan struktureras genom att göra en gemensam planering och rollfördelning. Vissa delar tar längre tid att göra och då kanske fler måste hjälpas åt med just det. Hur fördelas arbetet så att alla har något att göra genom hela projektet? Ett tips är att de som ska koda själva projektet jobbar med tillfälliga bakgrunder och sprites, medan de som ska rita eller designa spelet jobbar på vid sidan om. Sedan går det bra att ändra klädselar för att få in de egna spritarna.

INNAN PROGRAMMET

Övergripande syfte med avsnittet är att skapa en förståelse för att även skadlig kod är kod skapad av människor. Det är människor som skriver kod till virus och det är människor som skriver kod till viruskydd. Det absolut vanligaste sättet att hacka sig in i folks datorer på är genom att luras. Det kallas för social manipulation (engelska social engineering)

I avsnittet förekommer begreppen: tvåfaktorsautentisering, trojan, virus, mask, hacka, phishing,

UNDER PROGRAMMET

VISUALISERING

Inslaget visar olika typer av angrepp som kan drabba en på nätet. Det är människor som ligger bakom alla angrepp.

FRÅGA

Vad är phishing?

När man lockar någon att ge ut sina uppgifter. Kallas nätfiske på svenska.

UTMANING 1

Deltagarna ska beräkna säkerheten i lösenord. Med hjälp av en algoritm ska varje lösenord bedömas och sorteras i säkerhetsordning. Deltagarna har varken miniräknare eller papper och penna till hjälp och allt går på tid, så de har det lite stressigt att hinna med.

EXPERTEN

Avsnittets expert heter Siren Hofvander. Hon är Chief Security Officer på ett företag som hanterar mycket känslig information. Ett sätt att hålla sig extra säker på nätet är att använda tvåfaktorsautentisering. Det betyder att man använder två olika sätt för att logga in på en tjänst. Till exempel lösenord + ett sms. Siren påminner om att det inte finns någon raderaknapp på nätet, så man måste tänka efter innan man publicerar något. Sedan rekommenderar Siren alla att skaffa sig en lösenordshanterare som håller reda på alla lösenord eftersom man ska ha unika lösenord för alla tjänster.

UTMANING 2

För att kunna skydda sin information på datorn kan man använda kryptering. Då kan ingen som inte har rätt krypteringsnyckel se vad som finns på datorn. Det finns väldigt många olika sätt att kryptera på. Här har vi använt ett klassiskt ceasarkrypto för att göra orden svårslästa. Deltagarna har tre olika nycklar att välja mellan, för att lösa vilka de fyra krypterade orden är.

QUIZ

Vad är ett datorvirus gjort av?

Det är programmerad kod.

Vad heter det när man lurar någon att skriva in lösenord eller kontouppgifter?

Phishing eller nätfiske

Vad heter de recept som vi skriver till datorn?

Algoritm

Vad innebär det att något är krypterat?

Att det inte går att läsa om man inte har krypteringsnyckeln

EFTER PROGRAMMET

Här följer tips på övningar att göra efter programmet. Välj de uppgifter som passar gruppen. Titta gärna på Lektionstips som hör till avsnittet, för att se hur övningarna kan göras.

OBS! Övningarna som eleverna gör är inte rättade och ska inte användas som facit för vad som är rätt eller fel.

DISKUSSION

- Har ni råkat ut för hackare, virus eller något liknande på nätet?
- Social manipulation är det vanligaste sättet för hackare att ta sig in i folks datorer. Det behövs alltså ingen teknisk kunskap alls, utan bara smarta sätt att lura andra. Hur kan man skydda sig mot det?
- Går det att vara helt säker?
- Vad innebär det att vi är sårbara?
- Vad är det värsta som skulle kunna hända dig om någon obehörig tog sig in i din dator?

BERÄKNA SÄKERHET FÖR LÖSENORD

Kopieringsunderlag 6.1.1 och 6.1.2

Dela ut arbetsbladen. Listan visar Sveriges 25 vanligaste lösenord i september 2016. Observera att det är vuxnas lösenord det handlar om och att både knulla och kuken är med på listan. Det kan vara svårt att smälta för en del elever, men det är också en källa till en intressant diskussion. Om du känner att gruppens fokus kommer att handla för mycket om de enskilda orden – redigera listan.

Algoritmen för hur säkerheten ska bedömas kan göras på olika sätt. Uträkningen som görs i uppgiften är ingen absolut sanning. Det är inte perfekt uträknat att alla värden viktas rätt i formlerna. Det finns lite olika skolor för när lösenord anses säkra. En del menar att det bara är längden som avgör, medan andra menar att det är viktigt att blanda olika slags tecken. Absolut säkrast är om man använder helt slumpvisa lösenord från något slags lösenordsgenerator. Då behöver man också hjälp med att komma ihåg alla lösenord, eftersom man ska ha olika på alla tjänster man använder. En annan lösning är att ha en lång fras, mer än ett lösenord och att göra varje fras unik för den tjänst man loggar in på. "Ohvadunderbartattjagskaloggainpafacebooknu" skulle kanske vara en variant? Om den nu inte hade stått i den här lärarhandledningen förstås.

TVÅFAKTORSINLOGG

Kopieringsunderlag 6.2

Eftersom lösenord blir mindre och mindre säkert för varje år, rekommenderas vi använda tvåfaktorsautentisering när vi loggar in. Det betyder att det behövs två olika faktorer, alltså två olika sätt att bekräfta sin identitet, för att kunna logga in. De här faktorerna ska vara av olika typer: Något man ÄR, HAR och KAN. "Är" kan vara fingeravtryck eller ansikte. "Har" kan vara sms-kod som skickas till mobilen som man har. "Kan" är oftast ett lösenord.

Klipp isär de 14 orden på kopieringsunderlaget och hjälps åt att sortera dem efter vilken slags faktor de räknas som.

FACIT:

ÄR	HAR	KAN
ansiktsigenkänning	säkerhetsdosa	lösenord
fingeravtryck	SMS	svara på
röst	GPS	hemliga frågor
sätt att skriva	kodkort	PIN-kod
	ID-kort	

INNAN PROGRAMMET

Övergripande syfte med avsnittet är att förklara hur programmering styr de miljöer där vi tillbringar mycket tid om dagarna. Det är viktigt för att förstå digitaliseringens påverkan på samhället och för att främja ett kritiskt och ansvarsfullt förhållningssätt till sociala medier.

Fundera tillsammans: Vad är sociala medier? Vilka sociala medier använder eleverna?

I avsnittet förekommer begreppen: algoritm, sociala medier

UNDER PROGRAMMET

VISUALISERING

Inslaget visar hur Instagrams algoritm bestämmer i vilken ordning inlägg ska visas för olika användare.

FRÅGA

Vad kallas sorteringsrecepten i programmering?

Algoritm

UTMANING I VEM SER VAD?

I ett påhittat socialt nätverk har Karin och roboten Karim en del vänner som är gemensamma och en del som är helt skilda. Om Karin delar något med en vän som hon har gemensam med Karim, kommer han också att se det, även om det inte är meningen. Uppgiften för deltagarna är att lista ut vilka av Karins vänner som hon kan dela innehåll med, utan att Karim kan se det. Inte helt lätt det där, att hålla reda på vem som ser vad.

EXPERTEN

Avsnittets expert heter Janne Elvelid och han jobbar med sociala medier på ett av världens största företag, Facebook. Janne berättar hur viktigt han tycker det är med programmering och digital kompetens. Alla Facebooks tjänster byggs med programmering. Janne jobbar med att titta på hur till exempel lagarna ska anpassas bättre för att fungera i det digitala samhället.

UTMANING 2

Deltagarna ska programmera en robot att ta sig igenom en rätt besvärlig bana, utan att hamna i vattenhinder eller sandlåda. Roboten programmeras via en datorplatta, som sköts av en deltagare. Det är därför extra viktigt att gruppen kan samarbeta bra. Det finns två olika svängar, 45 och 90 grader. Det gäller att välja rätt.

QUIZ

Vad är sociala medier?

Kanaler där man kan prata med varandra, dela bilder och text.

Nämn ett sätt som sociala medier styrs av programmering på?

Sorterar bilder, styr vem som ser vilket innehåll.

Nämn några kända programmeringsspråk.

JavaScript, C++, C#, Python, PHP

Vad är en algoritm?

Ett recept för hur man ska lösa ett programmeringsproblem.

Vad är en slinga i programmering?

Ett annat ord för loop.

EFTER PROGRAMMET

Här följer tips på övningar att göra efter programmet. Välj de uppgifter som passar gruppen. Titta gärna på Lektionstips som hör till avsnittet, för att se hur övningarna kan göras.

OBS! Övningarna som eleverna gör är inte rättade och ska inte användas som facit för vad som är rätt eller fel.

DISKUSSION

- Vad tror ni kan ha förändrats i och med att vi började få sociala medier?
- Kan man välja bort alla sociala medier?
- Hur kan vi påverka hur vi beter oss på internet?
- Vad kan vi påverka med programmering?
- Vem/vilka är det som påverkar mest idag?

ANALYSERA SOCIALA MEDIER.

Kopieringsunderlag 7.1

I den här övningen är det en fördel om eleverna har egna konton i sociala medier som de kan analysera. Tanken är att de ska jobba i par eller i grupp och resonera kring frågeställningarna. Självklart kan man göra övningen tillsammans i helklass också.

- Anpassar sig tjänsten efter användaren? Hur? Exempel: Instagram, Facebook, Twitter och Youtube har alla algoritmer som gör att flödet anpassar sig efter användaren. Även om två personer följer exakt samma personer, kommer flödet att se olika ut eftersom de troligtvis gillar olika saker.
- Är tjänsten gjord så att användaren ska stanna kvar? På vilket sätt? I många tjänster presenteras förslag på andra konton att följa eller annat liknande material att titta på.
- Är tjänsten gjord så att innehållet är lätt att dela? Många tjänster har delaknappar så att inlägg lätt kan delas i andra sociala medier.
- Vill de som har gjort tjänsten att du ska dela innehåll med andra? Varför? Ja, för att fler ska komma till just den tjänsten. Instagram vill att Twitters användare ska komma till Instagram istället, och tvärtom. Fler användare ger mer reklamintäkter och fler användarkonton ger också mer tillgång till personliga data som värderas högt på marknaden.

DIGITOPIA

Kopieringsunderlag 7.2

Begreppet Digitopia kommer från den nederländske internetanalytikern Frank Buytendijk som använder det för att beskriva visioner om hur vi vill ha det digitala livet i framtiden. Men det som en tycker är den perfekta digitala framtiden stämmer kanske inte med vad du tycker. Vi har olika värderingar och när det kommer till vår samvaro på nätet är det nog ännu mer tydligt att människor har olika syn på hur det ska fungera.

Diskussionen om vilket internet vi vill ha i framtiden kan ske på olika sätt. I lektionstipset gjorde vi en värderingsövning där eleverna fick ställa sig på en linje i klassrummet. Om man istället vill ha svaren kvar kan man låta eleverna jobba i grupp och göra sin egen sammanställning över svaren på frågorna. Eller varför inte göra ett kollage med visionen för ett perfekt Digitopia? Kanske kan diskussionen mynna ut i en skrivövning där eleverna får beskriva en dag i sociala medier, så som det skulle kunna fungera i den bästa framtiden. Eller tvärtom, i en dystopi om hur samhället kan bli om vi inte passar oss?

AVSNITT 8 TEMA ROBOTAR

INNAN PROGRAMMET

Övergripande syfte med avsnittet är att förklara vad robotar är och på vilket sätt de har påverkat samhällsutvecklingen.

I avsnittet förekommer begreppen: robot, sensor, logisk grind, mjukvara, loop, slinga

UNDER PROGRAMMET

VISUALISERING

Inslaget visar robotarnas utveckling från 1495 fram till i dag. De första robotarna var mekaniska men inte programmerade. Den första industriroboten var en vävstol.

FRÅGA

Vad är skillnaden mellan en robot och en dator?

En robot utför ett fysiskt arbete.

UTMANING 1

Deltagarna ska sortera godis efter sex personers olika villkor. Villkoren illustreras med hjälp av logiska grindar.

EXPERTEN

Avsnittets expert heter Danica Kragic Jensfeldt. Hon är robotforskare på KTH i Stockholm och berättar om hur man måste studera människor för att kunna programmera robotar på ett bra sätt.

UTMANING 2

Deltagarna ska programmera två robotar att dansa, utan att krocka eller hamna utanför mattan.

QUIZ

När ritades den första människoliknanden roboten?

1495 alltså för ungefär 600 år sedan.

Inom vilken bransch började man tidigt att använda robotar i sina fabriker? *I textilindustrin där vävstolen var en av de första produktionsredskapen som ersattes av robotar.*

Vad betyder den här?  OCH

Vad betyder den här?  EJ

Är en diskmaskin en robot? Ja

Är en spis en robot? Nej

EFTER PROGRAMMET

Här följer tips på övningar att göra efter programmet. Välj de uppgifter som passar gruppen. Titta gärna på Programmera mera - Lektionstips som hör till avsnittet, för att se hur övningarna kan göras.

OBS! Övningarna som eleverna gör är inte rättade och ska inte användas som facit för vad som är rätt eller fel.

DISKUSSION

- Vilken robot har betytt mest för utvecklingen i samhället?
- Vilka robotar kan vi inte vara utan idag?
- Vilka robotar kommer att komma i framtiden?
- Finns det nackdelar med ökad robotisering?
- Finns det en risk att robotarna tar över helt?

Tips för inspiration: Svenska Simone Giertz har blivit världskänd för de riktigt dåliga robotar som hon tillverkar. På sin Youtubekanal visar hon upp läppstiftsroboten, popcornroboten och andra ljuvliga skapelser. <https://www.youtube.com/channel/UC3KEoMzNz8eY-nwBC34RaKCQ>

DANSPROGRAMMERING

Kopieringsunderlag 8.1

Dansprogrammering går att göra på flera olika sätt. Med yngre elever brukar det räcka med symboler (finns i lärarhandledning till Programmera mera). Det här är ett sätt att göra det lite mer avancerat för äldre elever. Det är ett påhittat skriftligt programspråk. Ge eleverna utmaningar som passar. I övningen står det att de får använda 10 rader kod att gå från punkt A till B, men det är självklart fritt fram att improvisera.

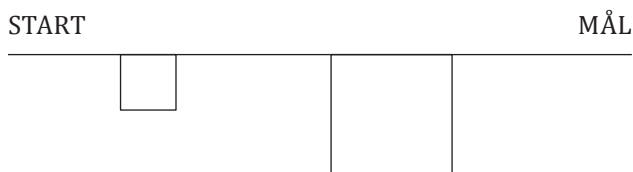
AMALIA	BÖRJE	CEASAR	DENA	EDDAN	FRANS
					
OCH /AND	EJ /NOT	ELLER / OR	ANTINGEN ELLER/ XOR	ELLER / OR	ANTINGEN ELLER/ XOR
Både skum och kvadrat.	Inte lakrits.	Gelé eller cirkel spelar ingen roll.	Antingen oval eller lakrits. Inte både delarna.	Kvadrat eller rektangel, spelar ingen roll.	Antingen skum eller gelé. Inte både delarna.

BOTMATTE

Kopieringsunderlag 8.2

Material som behövs är måttband och linjal och någon typ av robot, till exempel BeeBot eller BlueBot. Om skolan inte har möjligheter att köpa in robotar och det inte finns att låna eller hyra i närheten, så går det att använda appar och program på datorn, för att lära sig styra med hjälp av programmering. Det är inte nödvändigt att lägga ner tusentals kronor på robotar.

I lektionstipset ser vi hur eleverna får i uppgift att programmera sin robot att ta sig genom en bana där den ska gå rakt fram, sedan i en kvadrat och lite längre fram, en dubbelt så stor kvadrat, innan den går rakt i mål.



De får mäta hur långt deras robot tar sig på ett steg, men de får inte mäta hela banan, eftersom de ska träna rimlighetsbedömning vid uppskattningar.

De får också träna på att skapa en algoritm som kan återanvändas. I lektionstipset får eleverna använda en formel där X = antal, A = fram och B = sväng.

XA
X(XBXA)
XA
X(XBXA)
XA

Det skulle kunna vara:

2A
4(1B1A)
3A
4(1B2A)
4A

Genom att få eleverna att använda algebraiska uttryck och fundera på hur algoritmer kan skapas och användas vid programmering, så kan även en enkel robot som en beebot, användas av äldre elever.

Att styra

(Den här övningen finns inte i de lektionstips som hör till serien.)

I kursplanen för teknik finns i det centrala innehållet:

- Att styra egna konstruktioner eller andra föremål med programmering.

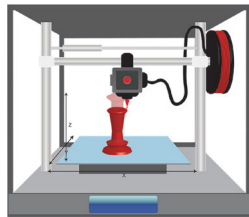
Det finns möjlighet att utforska det centrala innehållet i samband med robotavsnittet. Robotforskaren Danica Kragic Jensfeldt talar om sensorer hos robotar. Sensorer är ett sätt att styra robotar med. Programmera roboten att stanna eller svänga om den känner av ett hinder. Programmera roboten att följa en viss färg. Vissa sensorer kan reagera på ljus och ljud och kan styras den vägen istället.

OUTPUT

Klipp ut bilderna



SKRIVARE



3D-SKRIVARE



HÖGTALARE



SKÄRM



HÖRLURAR

BÅDE INPUT OCH OUTPUT



DATORPLATTA



HÖRLURAR
MED MIKROFON



BÄRBAR DATOR

INPUT



MIKROFON



DATORMUS



TANGETBORD



RITPLATTA

HEJ ALU/MINNE!
Ditt jobb som beräknare är att addera och subtrahera x och y när CU säger det till dig. Dessutom har du ett jobb som datorns minne att hålla reda på värdet på x och y , så att du kan säga det till CU när det behövs.

[illegible]

2

HEJ CU!

Ditt jobb som Central Processenhet (CU) är att köra följande program och säga åt de andra enheterna vad de ska göra.

- Mattekommandon som "Add 5 till x" eller "Sub 3 från y" skickas till ALU.
- Ritkommandon som "markera (x,y)" ska du skicka till Skärmen. Fråga först ALU efter gällande värden för x och y och ge dem sen till Skärmen.

Ibland är det bara värdet på x eller y som ändras. Då är värdet samma på den andra axeln. Se till att du markerar varje rad som du gör, så att du inte tappar bort var du är.

- | | | |
|---|---|---|
| ☐ add 5 till x | ☐ sub 3 från x | ☐ sub 3 från x |
| ☐ add 3 till y | ☐ add 4 till y | ☐ add 1 till y |
| ☐ markera (x, y) | ☐ markera (x, y) | ☐ markera (x, y) |
| ☐ add 1 till x | ☐ add 1 till x | ☐ sub 4 från y |
| ☐ add 3 till y | ☐ markera (x, y) | ☐ markera (x, y) |
| ☐ markera (x, y) | ☐ sub 2 från x | ☐ add 4 till x |
| ☐ sub 3 från x | ☐ markera (x, y) | ☐ markera (x, y) |
| ☐ sub 1 från y | ☐ sub 2 från x | ☐ sub 3 från x |
| ☐ markera (x, y) | ☐ sub 2 från y | ☐ add 1 till y |
| ☐ sub 2 från x | ☐ markera (x, y) | ☐ markera (x, y) |
| ☐ sub 2 från y | ☐ sub 1 från y | ☐ sub 2 från y |
| ☐ markera (x, y) | ☐ markera (x, y) | ☐ markera (x, y) |
| ☐ add 4 till x | ☐ add 6 till x | ☐ add 1 till x |
| ☐ sub 2 från y | ☐ markera (x, y) | ☐ markera (x, y) |
| ☐ markera (x, y) | ☐ add 1 till y | ☐ add 4 till y |
| ☐ add 2 till x | ☐ markera (x, y) | ☐ markera (x, y) |
| ☐ add 2 till y | ☐ sub 2 från x | |
| ☐ markera (x, y) | ☐ markera (x, y) | |

KLART!



Visa inte bilden för de andra förrän hela programmet är färdigt.

X

Y

	0	1	2	3	4	5	6	7	8	9
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Övningen är skapad av Gary Kacmarcik.

DOMÄNNAMNSSYSTEMET

Klipp ut siffrorna



domännamn	IP-adress
google.com	216.58.209.142
facebook.com	173.252.120.6
hitta.se	80.76.158.35
netflix.com	54.243.253.96
youtube.com	208.65.153.238
twitter.com	192.168.1.1
urskola.se	212.116.95.197

216.58.209.142

173.252.120.6

80.76.158.35

54.243.253.96

208.65.153.238

192.168.1.1

212.116.95.197

CAESARKRYPTO

klartext	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	Å	Ä	Ö
krypto																													

klartext	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	Å	Ä	Ö
krypto																													

klartext	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	Å	Ä	Ö
krypto																													

klartext	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	Å	Ä	Ö
krypto																													

klartext	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	Å	Ä	Ö
krypto																													

klartext	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	Å	Ä	Ö
krypto																													

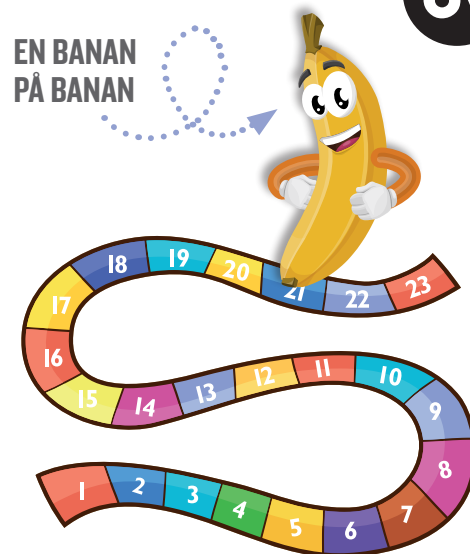
klartext	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	Å	Ä	Ö
krypto																													

VEM FÅR VILKET RESULTAT?

HOMOGRAFER – ord som stavas lika men uttalas olika

Ta fram två olika exempel på hemsidor.

Ett för varje betydelse för orden.



banan.....

bildrulle.....

fasan.....

fint.....

kort.....

kör.....

lounge.....

planet.....

svalt.....

tomten.....

JÄMFÖRA SÖKMOTORER

Här är några alternativ till google.
Prova att skriva in samma sökord i alla sökmotorer.

<https://www.bing.com/>
<https://www.duckduckgo.com/>
<https://www.sökis.se/>
<https://www.baidu.com/>
<https://www.yandex.com/>
<https://www.thinga.com/>



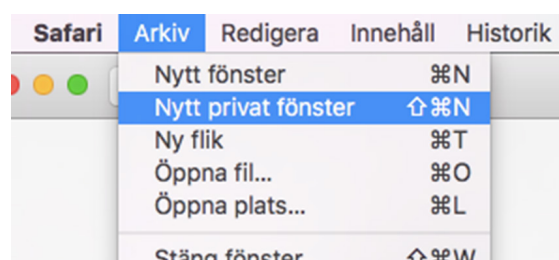
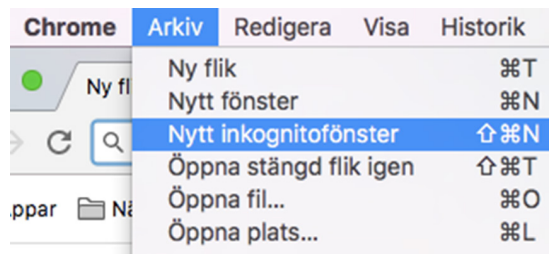
På vilka sätt skiljer sig resultaten?

.....

.....

.....

Ändra till privat fönster



Prova att söka igen. Ändras resultaten?

.....

.....

C++KOD – skriv gärna ut i A3

```

void Enemy::Reboot()
{
    isAlive = false;
    pTargetGameObject = NULL;
    pfnHitCallback = NULL;
    pfnMovementUpdate = NULL;
    Health = 1;
    HitTimer = 0;
    Timeout = 255;
}

void Enemy::SetDefinition( const EnemyDefinition* _pEnemyDefinition )
{
    if( pTargetGameObject == NULL )
    {
        pTargetGameObject = gameObjectManager.CreateGameObject( _pEnemyDefinition->pAnimationDefinition );

        pTargetGameObject->SetHotspot( 0, 0 );
        pTargetGameObject->m_flags = GO_FLAGS_ENEMY;
        pTargetGameObject->GetSprite()->collisionIndex = SPRITE_COLLISION_INDEX_ENEMY;
        pTargetGameObject->m_customObject = this;
        pTargetGameObject->m_customUpdate = enemyUpdatePipe;
    } else
    {
        pTargetGameObject->GetAnimation()->SetSequence( _pEnemyDefinition->pAnimationDefinition );
    }
    pTargetGameObject->GetAnimation()->Play();

    Health = _pEnemyDefinition->StartHealth;
    SpecialFlag = _pEnemyDefinition->SpecialFlag;

    HitTimer = 0;
    Timeout = 255;

    pfnMovementUpdate = _pEnemyDefinition->pfnMovement;

    isAlive = true;
}

void Enemy::Update()
{
    if( Timeout > 0 )
    {
        Timeout--;
        if( Timeout == 0 )
        {
            pTargetGameObject->SetEnabled( false );
            return;
        }
    }

    if( HitTimer > 0 )
    {
        HitTimer--;
        if( HitTimer == 0 )
        {
            pTargetGameObject->GetSprite()->flags &= ~SPRITE_FLAG_DRAWWHITE;
        }
    }
}

```

```

    if( AttackTimer > 0 )
    {
        AttackTimer--;
        if( AttackTimer == 0 )
        {
            // Attack the player

            fp2d targetPosition = fp2d( player->GetWorldPositionX()+20, player->GetWorldPositionY()+3 );

            enemyBulletSpawn( m_worldPosition.x.GetInteger(), m_worldPosition.y.GetInteger(), targetPosition, FixedPoint( 0, 50 ));
        }
    }
    if( pfnMovementUpdate != NULL )
    {
        pfnMovementUpdate( this );

        pTargetGameObject->SetWorldPosition( m_worldPosition.x.GetInteger(), m_worldPosition.y.GetInteger());
    }
}

bool Enemy::Hit()
{
    if( SpecialFlag == ENEMY_SPECIALFLAG_INVINCIBLE ) return false;

    if( SpecialFlag == ENEMY_SPECIALFLAG_SPACEBASEBOSSDOOR ) return false;
    if( Health <= 1 )
        return true;

    pTargetGameObject->GetSprite()->flags |= SPRITE_FLAG_DRAWWHITE;

    Health--;
    HitTimer = 3;

    if( pfnHitCallback != NULL )
        pfnHitCallback( this );
    return false;
}

void Enemy::Kill()
{
    if( SpecialFlag == ENEMY_SPECIALFLAG_SPACEBASEBOSSDOOR )
        return;
    pTargetGameObject->SetEnabled( false );
}

void Enemy::SetWorldPosition( const fp2d& _worldPosition )
{
    m_worldPosition = _worldPosition;
    pTargetGameObject->SetWorldPosition( m_worldPosition.x.GetInteger(), m_worldPosition.y.GetInteger());
}

void Enemy::SetWorldPosition( int _x, int _y )
{
    m_worldPosition.x = _x;
    m_worldPosition.y = _y;
    pTargetGameObject->SetWorldPosition( _x, _y );
}

```

STORYBOARD TILL SPEL

Bakgrunder:			
Sprites:			

Huvudansvar programmering	
Huvudansvar för utseende bakgrund	
Huvudansvar för utseende Sprites	
Huvudansvar storyboard	
Huvudansvar Textinnehåll	

SVERIGES VANLIGASTE LÖSENORD 2017

	Säkerhetspoäng
1. 123456	
2. 12345	
3. knulla	
4. qwerty	
5. 666666	
6. hejsan	
7. stilet	
8. 123456789	
9. 12345678'	
10. password	
11. mamma	
12. volvo	
13. sommar	
14. hejhej	
15. niklas	
16. bajskorv	
17. gnaget	
18. artto	
19. kungen	
20. kalle	
21. sverige	
22. general	
23. kalleanka	
24. kuken	
25. 123123	

BERÄKNA SÄKERHET FÖR LÖSENORD

Det finns olika sätt att beräkna hur säkert ett lösenord är. Det här är en algoritm som bedömer lösenorden efter längd och variation. Börja med att räkna ut alla pluspoäng, och ta sedan alla minus. Ju högre tal, desto säkrare lösenord.

LÄNGD: $+(\text{antal tecken} \times 4)$

STORA BOKSTÄVER: $+(\text{ordets längd} - \text{antal stora bokstäver}) \times 2$

SMÅ BOKSTÄVER: $+(\text{ordets längd} - \text{antal små bokstäver}) \times 2$

SIFFROR: $+(\text{antal siffror} \times 4)$

SYMBOLER: $+(\text{antalet symboler} \times 6)$

BARA BOKSTÄVER: $-\text{antal bokstäver}$

BARA SIFFROR: $-\text{antal siffror}$

BARA STORA BOKSTÄVER: $-(\text{antal stora bokstäver} \times 2)$

BARA SMÅ BOKSTÄVER: $-(\text{antal små bokstäver} \times 2)$

BOKSTÄVER I ALFABETISK ORDNING: $-(\text{antal bokstäver i ordning} \times 3)$

SIFFROR I ORDNING: $-(\text{antal siffror} \times 3)$

ORD I ORDLISTA: $-(\text{antal bokstäver} \times 4)$



Illustration of a login form with two input fields and a button. The first field is labeled 'användarnamn' (username) and has a person icon. The second field is for a password, indicated by a lock icon and masked with asterisks. Below the fields is a 'LOGGA IN' (Log In) button.

TVÅFAKTORSINLOGG

Klipp ut orden



ansiktsgenkänning	ögonskanning
fingeravtryck	GPS
lösenord	PIN-kod
Svara på hemliga frågor.	ID-kort
säkerhetsdosa	kodkort
SMS	Sätt att skriva på tangenter.
röst	USB-sticka

ANALYSERA SOCIALA MEDIER

Välj en tjänst som ni använder ofta

Gå till sidan. Om någon av er har ett konto, logga in



1) Anpassar sig tjänsten efter användaren? Hur?

.....

2) Är tjänsten gjord så att användaren ska stanna kvar? På vilket sätt?

.....

3) Är tjänsten gjord så att innehållet är lätt att dela?

.....

4) Vill de som har gjort tjänsten att du ska dela innehåll med andra? Varför?

.....

.....

.....

Internet är i grunden helt öppet för alla. Det heter att det är transparent. Att alla kan se vad andra gör. Sociala medier är en del av det. Vi ska dela allt med alla. Hur vill du ha det? Ska allt vara transparent eller allt anonymt?

Det finns teknik som gör att internetleverantörer kan begränsa vilka tjänster som kan användas. Är det bra att det finns, eller borde allt vara fritt?

Inget som finns på internet försvinner, sägs det ibland. Är det bra att allt lagras, eller ska man ha rätt att ta bort saker från nätet? Om man kan ta bort innehåll, ska alla få ta bort det då? Politiker, lärare, brottslingar

VILKA SKA HA RÄTT ATT VARA ANONYMA?

1. brandman
 2. deltagare i demonstrationståg
 3. fotbollssupporter
 4. polis
 5. lärare
 6. läkare
 7. journalist
 8. fritidsledare
 9. popstjärna
 10. politiker
 11. vittne vid rättegång
- 

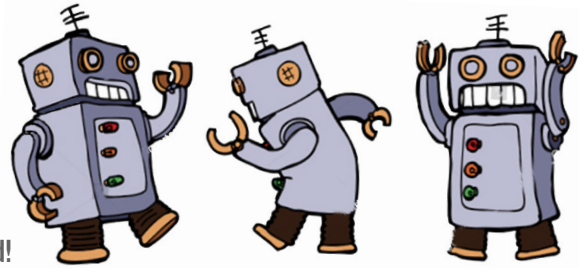


DANSPROGRAMMERING

Utmaning:

Programmera en robotdans som flyttar din robot från A till B.

Programmet skapar du genom att välja 10 av nedanstående kommandon och funktioner/metoder. Om roboten inte tar sig till B får du gärna felsöka, justera programmet och testa igen tills du är nöjd!



[kommandon]

Specifika instruktioner för att styra robotens armar, ben, huvud, händer etc

robot.3pek:hoArm Dansa 3 pekdisco med höger arm
 robot.3pek:vaArm Dansa 3 pekdisco med vänster arm
 robot.3rull:Arm..... Dansa 3 varv rulldisco med båda armarna
 robot.3kick:hoBen Kicka 3 ggr åt höger
 robot.3kick:vaBen..... Kicka 3 ggr åt vänster
 robot.3:framBen..... Ta 3 steg framåt
 robot.3:bakBen Ta 3 steg bakåt
 robot.3:hoBen Ta 3 steg åt höger
 robot.3:vaBen Ta 3 steg åt vänster
 robot.90ho:Body Vrid till höger (90 grader)
 robot.90va:Body Vrid till vänster (90 grader)
 robot.3stamp:hoBen Stampa 3 ggr med höger fot
 robot.3stamp:vaBen Stampa 3 ggr med vänster fot
 robot.3knapp:hoArm Knäpp 3 ggr med höger hands fingrar
 robot.3knapp:vaArm Knäpp 3 ggr med vänster hands fingrar
 robot.360snurr:Body Snurra 1 varv (360 grader)
 robot.2repeat:Next..... Upprepa förra raden. Efter 2 ggr gå till nästa rad

[funktioner och metoder]

Anropa färdiga sekvenser som roboten redan vet hur den ska utföra

robot.gangnam:Body..... Gangnam style
 robot.stirpot:Body..... Rör om i grytan
 robot.roof:Body Händerna i taket
 robot.dab:Body Dab
 robot.twist:Body Enkel twist

Övningen är skapad av Terese Raymond, Malin Ströman och Anders Ekbom.

BEEBOTMATTE

1. Mät hur långt boten går på ett tryck.
2. Uppskatta avståndet på banan.
3. Programmera boten att ta sig från start till mål och undvika hindren på vägen, utan att testa i banan.

4. Skriv ner koden enligt formeln:

X = antal

A = fram, B = höger, C = vänster, D = bakåt, C = paus

XA

X(XBXA) (det här föreställer en loop. Det går inte att trycka in på boten, men ni kan skriva loopen i koden ändå)

XA

X(XBXA)

XA

5. Testa banan tillsammans.
6. Hur gick det?
7. Hade ni kunnat göra något annorlunda?