

# Lärarhandledning

## Lär dig logik med VR – Logic Gates -

### Logic Gates VR

#### Vad är Lär dig logik med VR – Logic Gates?

En VR-upplevelse där eleverna tränar på logiska portar (logic gates) genom att koppla "sladdar" och bygga logiska uttryck steg för steg. När kopplingen är korrekt tänds en lampa som visar att slutresultatet blir sant (1) eller falskt (0). Upplevelsen fungerar bäst som fördjupning för elever som redan kommit igång med programmering och grundläggande villkorsstyrning.

#### Syfte

- Fördjupa elevernas förståelse av logik som grund för programmering och digital teknik.
- Träna eleverna i att bygga, testa och felsöka logiska villkor utöver AND/OR/NOT på ett konkret och visuellt sätt.
- Skapa broar mellan matematikens logiska resonemang och teknikens elektroniska styrning och reglering.

#### Material

- VR-headset (Meta Quest eller liknande)
- Meta-konto för nedladdning av appen
- Appen *Logic Gates VR* installerad

Behöver du hjälp eller söker mer information om Meta-konto, konfigurering och installation av appar kan du läsa mer här: [Meta Store Hjälpcenter](#)

#### Rekommenderad åldersgräns

- Rekommenderas främst för åk 7–9, särskilt elever som:
- kan grundläggande programmering och villkor, men fastnar på logiken
- behöver fler logiska villkor utöver AND/OR/NOT och hur de kombineras
- För nybörjare i programmering är det ofta bättre att först arbeta analogt med begreppen IF, ELSE, AND, eller, NOT innan man går vidare till portar och mer formell logik.

#### Fysisk säkerhet och utrymme

- **Tillräcklig yta:** Se till att varje elev har 2 × 2 meter fri yta att röra sig på.
- **Fri golvyta:** Rensa bort mattor, sladdar eller andra saker som elever kan snubbla över.
- **Sittande alternativ:** Upplevelsen kan genomföras sittande för att minska fallrisk och göra det lättare att fokusera på skapandet.

#### Övervakning och stöd

- **Pedagogisk närvaro:** En vuxen bör alltid finnas närvarande för att vägleda och ingripa vid behov.

## Hälsa och välmående

- **Tidsbegränsning:** Håll passet kort per elev för att undvika ögonbelastning. Lägg in mikropausar mellan elever.
- **Känslighet för rörelsesjuka:** Risken för yrsel är låg, men låt elever avbryta när som helst om de känner sig obekväma.

## Hygien och utrustning

- **Rengöring:** Desinficera VR-headset (delen som är mot huden) och kontroller mellan användning, särskilt om flera elever delar utrustningen. Behöver linserna rengöras, använd endast en mikrofiberduk (samma sort som används till glasögon).

OBS! Använd INTE rengöringsmedel mot linserna då vissa rengöringsmedel kan förstöra linserna.

## Läroplanskoppling (LGR22)

### Centralt innehåll i Matematik för årskurs 7-9

- Programmering i visuell och textbaserad programmeringsmiljö. Hur algoritmer skapas, testas och förbättras vid programmering.

### Centralt innehåll i Teknik för årskurs 7-9

- Hur komponenter och delsystem benämns och samverkar inom tekniska system, till exempel informations- och kommunikationsteknik och transportsystem.
- Tekniska lösningar för styrning och reglering med hjälp av elektronik och olika typer av sensorer. Hur tekniska lösningar som utnyttjar elektronik kan programmeras samt begrepp som används i samband med detta.
- Egna konstruktioner där man använder styrning eller reglering med hjälp av programmering

### Didaktisk kommentar

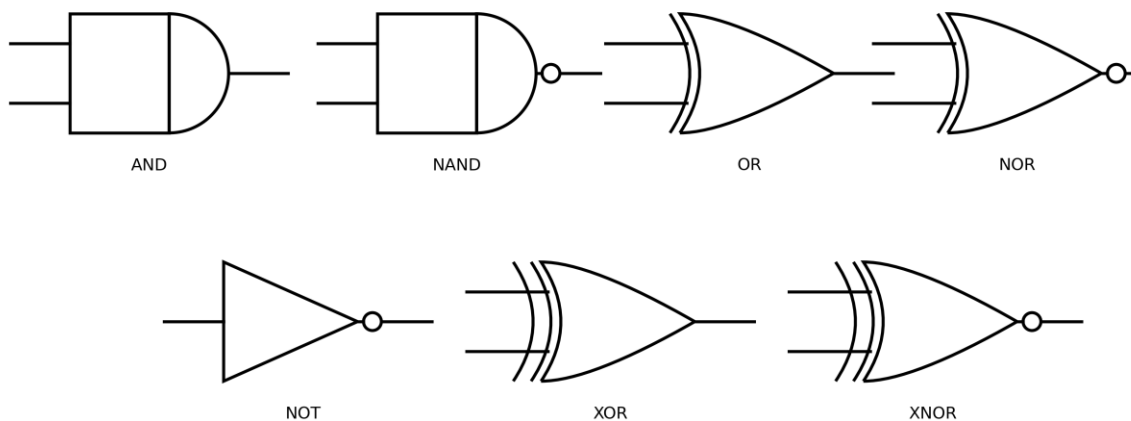
Även om "logik" inte alltid nämns explicit i kommentarsmaterialet till kursplanen i matematik eller kursplanen i matematik/teknik är logiska villkor i praktiken en nödvändig del av programmering (till exempel i scratch och i kalkylprogram), där elever behöver hantera uttryck med IF, AND, eller och ELSE. För att kunna göra mer avancerade program behövs fler logiska villkor.

Observera att denna upplevelse är en fördjupning för elever som är intresserade av programmering.

## Genomförande

- Repetera om, och, eller, inte och annars med 2–3 snabba exempel på tavlan.
- Visa symbolerna för AND/OR/NOT och låt eleverna gissa utfallet för några ingångar.
- Förklara målet i VR: Bygg en koppling som gör att slutlampan tänds (rätt logiskt resultat).
- Differentiering: Bas – bara AND/OR/NOT. Fördjupning – NAND/NOR/XOR/XNOR.

## Logikportar (standard-symboler)



Förklaring: ◦ (bubbla) = INTE (NOT). XOR har en extra böjd linje vid ingången.

### Reflektion och avslutning

- Vilken port var lättast/svårast – och varför?
- Gör en sanningstabell
- Vad var det vanligaste felet: fel port, fel ordning, eller glömt NOT?

### Vidare arbete

- Låt eleverna bygga samma logik i Scratch (villkorsblock) eller i en textbaserad miljö, kalkylprogram kan också använda dessa villkor.
- Gör en karta över en liten villa med larmsensorer. Gör pseudokod
- Om larm aktiverat OCH rörelsesensor 1 Aktiverad så sätt inbrottslarm till 1  

$$\text{LARM} = 1 \text{ if } (\text{larm\_aktiverad} == 1 \text{ and } \text{rorelsesensor\_aktiverad} == 1) \text{ else } 0$$

## Begreppslista

Logiken bygger på att en slutsats blir sann (1) eller falsk (0) beroende på olika villkor

### Grundportar

AND – utgången blir 1 bara om båda ingångarna är 1.

OR – utgången blir 1 om minst en ingång är 1.

NOT – inverterar: 1 blir 0, 0 blir 1 (har en ingång).

### Vanliga portar för fördjupning

NAND – NOT(AND). Vanlig i digital elektronik; man kan bygga andra portar av NAND (motsats till AND).

NOR – NOT(OR). Kan också användas för att bygga andra portar. (motsats till OR)

XOR – 1 om ingångarna är olika (0/1 eller 1/0). (Enbart någon är korrekt men aldrig fler)

XNOR – 1 om ingångarna är lika. (Motsats till XOR)