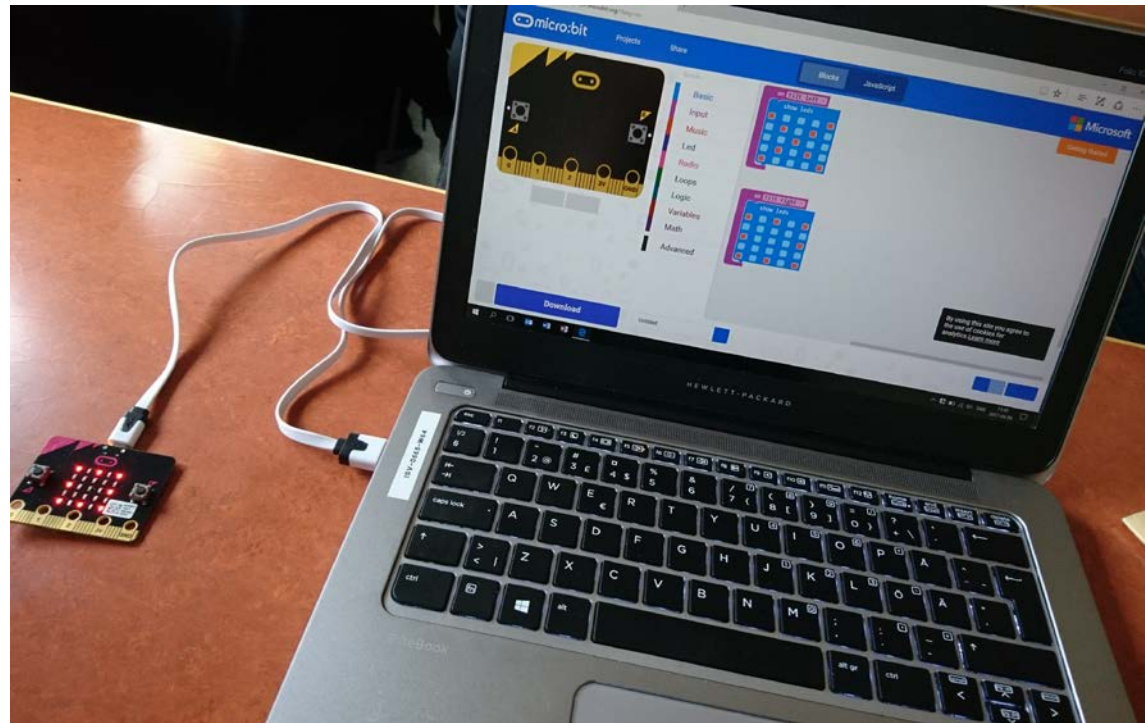


# Vad säger forskningen om programmering som kunskapsinnehåll?

Karin Stolpe, föreståndare NATDID  
[liu.se/natdid](http://liu.se/natdid)

# Programmering i skolan



# Lgr 11 (rev. 2017)

## ***Arbetsätt för utveckling av tekniska lösningar***

### **Åk F-3**

Att styra föremål med programmering.

### **Åk 4-6**

Att styra egna konstruktioner eller andra föremål med programmering.

### **Åk 7-9**

Egna konstruktioner där man tillämpar styrning och reglering, bland annat med hjälp av programmering.

## ***Tekniska lösningar***

### **Åk 7-9**

Tekniska lösningar som utnyttjar elektronik och hur de kan programmeras.

Utbildningen ska vila på vetenskaplig grund och beprövad erfarenhet.  
(Skollag 2010:800)



# Vad säger forskningen?

Du blir bra på det du övar!

J. EDUCATIONAL COMPUTING RESEARCH, Vol. 51(2) 185-204, 2014



Journal of Educational Computing Research on Technology in Education



**EFFECTS OF USING ALICE AND SCRATCH IN AN INTRODUCTORY PROGRAMMING COURSE FOR CORRECTIVE INSTRUCTION**

18 (Online) Journal homepage

Computers in Human Behavior 52 (2015) 200-210

Contents lists available at ScienceDirect

**Computers in Human Behavior**

journal homepage: [www.elsevier.com/locate/comphumbeh](http://www.elsevier.com/locate/comphumbeh)



**What  
manij**



Lou Sla

**ELSEVIER**

**A new way of teaching programming skills to K-12 students: Code.org**  
Filiz Kalelioğlu  
Baskent University, Faculty of Education, Ankara, Turkey

**"Monkey-Pokey!":  
Learning Approaches and Achievement  
Throughout Early Cognitive Development**

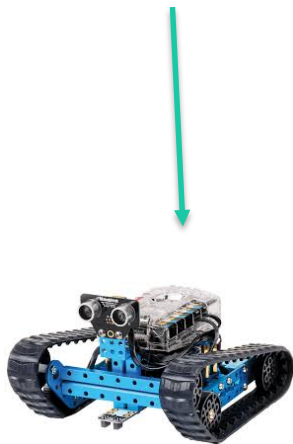
Louise P. Flannery & Marina Umaschi Bers

Tufts University



Achievement  
Development  
[ste.org/jrte](http://ste.org/jrte)

Elever som får arbeta  
med programmering  
blir duktigare på att  
programmera.



```
File Edit Format Run Options Windows Help
# Hello.
# Creating a Vending Machine Simulation in Python:

def vendingMachine ():
    totalCoins = 0
    coins = 0
    finished = False
    acceptableCoins = [0,10,20,50,100]
    while finished == False:
        try:
            coins = int (input ("Please enter your coins: "))
            while coins not in acceptableCoins:
                print ("Sorry, this input is not valid.")
                print ("Only 10p, 20p, 50p and 100p are accepted.")
            coins = int (input ("Please enter your coins: "))
            totalCoins = coins + totalCoins
        except ValueError:
            print ("The value you have entered is not acceptable.")
            print ("Please try again.")
            coins = int (input ("Please enter your coins: "))
            while coins not in acceptableCoins:
                print ("Sorry, this input is not valid.")
                print ("Only 10p, 20p, 50p and 100p are accepted.")
            coins = int (input ("Please enter your coins: "))
            totalCoins = coins + totalCoins
    if coins == 0:
        finished = True
    # Now we add the 'acceptance' bit:
    print ("Your input has been accepted.")
    print ("You have {0}p in your bank.".format (totalCoins))
    print ("You now need to choose your product(s)")
    print ("")
```

# Men...

...effekter på elevers  
**datalogiska tänkande** är  
generellt sett marginella.

**Datalogiskt tänkande**  
används som argument för  
att införa programmering i  
skolan.



“Computational thinking is the *thought process* involved in formulating a problem and expressing its solution(s) in such a way that a computer – human or machine – can effectively carry it out.” (Wing, 2017)

Problemlösning

Abstrakt tänkande

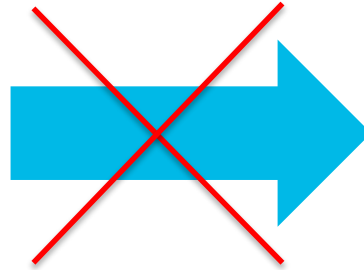
Kreativitet

Systematisk felsökning

Logiskt tänkande

Förmåga att se mönster

**Programmering**



**Datalogiskt tänkande**

## Samtidigt...

Beroende på vad som är i fokus, kan eleverna lära sig andra saker genom att programmera.

**11-12 år**

**Uppgift:** Designa, programmera och bygg en robot som själv kan hitta till en "ö", utan att köra in i saker, stanna på ön och höja en flagga.



# För att lösa problemet

- Se systemet som helhet
- Identifiera olika undersystem med delar, processer och hur de samverkar



Ljussensorn måste  
kanske sitta mellan

Som det är nu så  
kommer den att

fä  
[rol

Men han har  
fortfarande inte  
kommit fram till ön.



Varför då?

Varför är det en  
nackdel?

# Kan vi lita på resultaten?

- Fler studier visar på liknande resultat (Sullivan & Heffernan, 2016).

Det är viktigt att bestämma vad man vill att eleverna ska lära sig.



# Lärarens roll

- Läraren är viktig
- Effektivisera genom att t.ex. använda loopar



# Blockprogrammering eller textbaserad programmering?

Block-  
programmering

Text-  
programmering

Test 1: Blockprogrammering och textprogrammering

Text-  
programmering

Block-  
programmering

Test 2: Blockprogrammering och textprogrammering

# Så vad lär vi oss av detta?

## Programmeringsdidaktik

HUR?

VAD?

VARFÖR?

Skynda långsamt...

Mer forskning behövs...

Kritiskt förhållningssätt



## Referenser i urval

- Chao, P.-Y. (2016). Exploring students' computational practice, design and performance of problem-solving through a visual programming environment. *Computers & Education* 95: 202-215.
- Fessakis, G., Gouli, E., & Mavroudi, E. (2013). Problem solving by 5-6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*, 63, 87-97.
- Harlow, D. B., & Leak, A. E. (2014). Mapping students' ideas to understand learning in a collaborative programming environment. *Computer Science Education*, 24(2-3), 229-247.
- Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*. 52: 200-210.
- Kalelioğlu, F. & Gülbahar, Y. (2014). The effects of teaching programming via Scratch on Problem Solving Skills: A discussion from learners' perspective. *Informatics in Education*. 13(1): 33-50.
- Okita, S. Y. (2014). The relative merits of transparency: Investigating situations that support the use of robotics in developing student learning adaptability across virtual and physical computing platforms. *British Journal of Educational Technology*, 45(5), 844-862.
- Scherer, R. (2016). Learning from the past – the need for empirical evidence on the transfer effects of computer programming skills. *Frontiers in Psychology*. 7:1390.
- Slangen, L., van Keulen, H., & Gravemeijer, K. (2011). What pupils can learn from working with robotic direct manipulation environments. *International Journal of Technology and Design Education*, 21(4), 449-469.
- Straw, S., Bamford, S., & Styles, B. (2017). *Randomised controlled trial and process evaluation of code clubs*. Slough: NFER.
- Sullivan, F. R., & Heffernan, J. (2016). Robotic Construction Kits as Computational Manipulatives for Learning in the STEM Disciplines. *Journal of Research on Technology in Education*, 48(2), 105-128.
- Wing, J. M. (2017). *Computational Thinking*. Presentation at the Trippel Helix Conference on Computational Thinking and Digital Competences in Primary and Secondary Education, Stockholm.

[www.liu.se/natdid](http://www.liu.se/natdid)  
[karin.stolpe@liu.se](mailto:karin.stolpe@liu.se)

[www.liu.se](http://www.liu.se)