



Programmering som ett verktyg för att lösa matematiska problem

Andreas Borg

Karlstads universitet (Sundsta-Älvkullegymnasiet)

Resultat

Studien, där gymnasieelever ombads lösa matematiska problem med hjälp av programmering, visade dels hur eleverna initialt hade svårt att inse de matematiska möjligheter som en programmeringsmiljö potentiellt kan erbjuda men också hur övergången från matematiska notationer till kod orsakade många elever bekymmer. Dessutom upplevde flera elever svårigheter kopplat till användningen av datalogiska iterationer (loopar).

Slutsats

Enbart programmeringskunskap är inte tillräckligt för att elever ska kunna använda en programmeringsmiljö som ett matematiskt verktyg. Eleverna måste därtill ges tid att anpassa programmeringsmiljön för matematiska syften. Programmering kan dock erbjuda eleverna nya matematiska Lösningsstrategier samt konkretisera matematiska begrepp såsom variabelvillkor.

Bakgrund

Studien tog sin utgångspunkt i den reviderade ämnesplanen i matematik som togs i bruk 2018 och som fastslog att programmering, i vissa gymnasiekurser, skulle användas som ett verktyg vid matematiskt problemlösning. Denna förändring ligger i linje med en internationell trend där kunskap om programmering ses som en viktig del av en allmän digital kompetens.

Metod

I studien användes designforskning som metodologisk ansats. Designforskning är en cyklisk (se bild nedan) och praktisknära metodologisk ansats där forskningen bedrivs i klassrummet i vilket exempelvis en designad lärandesituation testas för att sedan analyseras och utvärderas utifrån uppställda lärandemål. Denna analys leder sedan fram till slutsatser samt förslag på en reviderad design inför nästkommande designcykel.



Kontaktuppgifter

andreas.borg@kau.se

0705 20 64 66

```
5 public static void main(String[] args) {
6
7     int a;
8     for(a = 0; a < 1000; a++){
9         int b = a + 11;
10        for(int f = a; f < b; f++){
11            if ((b+1)*(f+1)*(a+1) - (b+1)*(f+1)*a == 432) {
12                break;
13            }
14        }
15    }
16    System.out.print(a);
17 }
```

