

Vad är en Flowchart?

En flowchart är en typ av diagram som representerar en algoritm, arbetsflöde eller en process. Med en flowchart kan vi visa ett programs förväntade exekvering, alltså hur det ska bete sig, med hjälp av olika symboler. Dessa symboler kopplas sedan ihop med pilar för att visa hur datan flödar från en komponent till en annan.

Varför en flowchart?

Flowcharts används för att designa och dokumentera processerna i ett program. Med hjälp av en flowchart kan vi skapa en ritning som beskriver hur programmet ska bete sig eller vad det förväntas göra. Detta ger oss också en överblick på hur vi ska strukturera upp ett program innan dess att vi börjar koda det vilket förebygger olika typer av fel. Nedan följer ett exempel på hur en flowchart för bevattning av en blomma kan se ut, tillsammans med motsvarande javascript.

Flowchart	JavaScript
<pre>graph TD; Start([Start]) --> Fill[Fill water can]; Fill --> Moist{Is plant soil moist?}; Moist -- Yes --> Done([Done]); Moist -- No --> Succulent{Is plant a succulent?}; Succulent -- Yes --> Some[Add some water]; Succulent -- No --> Lots[Add lots of water]; Some --> Done; Lots --> Done;</pre>	<pre>fillWaterCan(); if(plantSoilMoist == true) { done; } else { If(plantASucculent == true) { addSomeWater(); done; } else { addLotsOfWaer(); done; } }</pre>

De två vanligaste symbolerna är
en process/aktivitet – den rektangulära boxen
ett beslut – diamanten

Ni kan läsa mer om de olika symbolerna och dess betydelse på följande länk

<http://creately.com/diagram-type/article/introduction-flowchart-symbols>

Vad är pseudokod?

Pseudokod är ett verktyg som används för att prototypa och dokumentera vår kod. Istället för att vi skapar programmet i ett valt programmeringsspråk, t.ex. javascript, använder vi det engelska språket för att skriva instruktioner som programmet ska utföra. Dessa instruktioner kan sedan logiskt utvärderas och översättas av en programmerare till ett visst programmeringsspråk. Pseudokod är precis som dataflows en metod för att designa en process/algorithm.

Pseudokod har ingen bestämd form och det finns inga strikta regler för hur den bör skrivas. Det viktigaste är att koden är en kombination av programmeringens logik och ett språk som programmeraren är bekant med, t.ex. engelska. Genom att skriva pseudokod istället för vanlig kod kan programmeraren förstå vad en algoritm gör utan att kunna det specifika programmeringsspråket den är skriven för.

Varför pseudokod?

När vi försöker lösa ett problem inom programmering är det inte alltid lätt att komma igång med den riktiga koden direkt. Ibland vet vi hur vi ska lösa ett problem, men inte riktigt hur vi ska formulera vår kod för att göra det. Detta gör att vi både blir frustrerade och kastar bort värdefull tid genom att programmera utan egentligen ha någon struktur på vad vi ska göra.

Pseudokod gör skapandet av program enklare. Med hjälp av pseudokod kan vi analysera vår lösning och faställa effekten av en algoritm innan dess att vi börjar koda med ett programmeringsspråk, förberedelse är viktigt. Vi undviker helt enkelt att koda fram en lösning i ett specifikt språk som i slutändan kanske inte löser problemet.”

Hur skriver jag pseudokod?

Allt du behöver kunna för att skriva pseudokod är det engelska språket. Vi skriver vart vårt program ska göra på engelska, rad för rad, precis som vanlig programmering. Du använder dina egna ord och instruktioner för att skapa en struktur. I exemplet nedanför vill jag göra ett program som lägger ihop två siffror som en användare skriver in, och sedan visar resultatet för användaren. Till vänster ser ni pseudokoden vi skriver först, och till höger ser ni översättningen i javascript.

<i>Pseudokod</i>	<i>JavaScript</i>
<i>Start program Enter two numbers A, B Add the numbers together Print the sum of the numbers End program</i>	<i>var a = prompt("Enter first number"); var b = prompt("Enter second number"); var c = (a+b); alert("Summan är " + c);</i>

Kom ihåg att pseudokod inte är något som kan läsas av en dator och kommer aldrig att kompileras till ett program. Det är avsett för oss människor att själva läsa och översätta till brukbar kod. Du kan skriva pseudokod precis hur du vill så länge andra förstår den. Vissa företag har dock en specifik syntax på sin pseudokod så att alla skriver liknande kod. Syntax är regler för hur kod skrivs och organizeras i ett språk och parenteser samt klamrar är en del av syntaxen i JavaScript.

r ser ni en flowchart för som motsvarar pseudokoden ovanför

