

Viktiga begrepp

Den här ordlistan är till för dig som går kursen Om Programmering. Eftersom detta är en grundläggande kurs har vi i vissa fall gjort en del förenklingar. En del begrepp är svåra att förenkla, men du behöver inte kunna alla för att klara kursen. Stöter du på ett begrepp du inte förstår, hoppa över det och läs vidare.

Algoritm

En algoritm är en begränsad samling väldefinierade instruktioner för att lösa en specifik uppgift med ett begränsat antal steg. En algoritm beskrivs oftast i vanlig text som stegvisa instruktioner i och implementeras i en dator med hjälp av programmering.

Skillnaden mellan en algoritm och ett program är att algoritmen inte är skriven i ett specifikt programspråk utan istället bara beskriver tillvägagångssättet. Samma algoritm kan realiseras i olika programspråk och i olika program.

Ett matrecept kan sägas vara en enkel form av algoritm. Inom programmering kan en algoritm exempelvis användas för att sortera en samling tal i storleksordning, automatiskt justera ventilation, anpassa priser i en affär baserat på efterfrågan eller bestämma vad du får se i ditt flöde i sociala media.

Array

Se vektor.

Binärkod

Se maskinkod.

Blockprogrammering

Blockprogrammering, eller programmering i visuella programmeringsmiljöer, är en variant av programmering där programmeraren använder grafiska element för att bygga upp sitt program. Används primärt i utbildningssyfte men är också populärt i t.ex. vissa programmeringsmiljöer för dataspel, arkitektur, musik och enklare kontrollsystem. Blockprogrammering används också för att bygga upp grafiska gränssnitt till program.

Bugg / fel och felsökning

En bugg är ett fel i ett program som gör att programmet inte fungerar som det är tänkt. Ibland kan en bugg betyda att programmet fortsätter köra men ger ett felaktigt svar (t.ex. vid en beräkning) och i andra fall kan det betyda att programmet kraschar, dvs. slutar köra.

Ordet "bugg" (från engelskans bug som betyder insekt) sägs komma ifrån att en insekt, en mal flög in i en stordator som kraschade. Detta skulle ha hänt 1947.

Dataspel

Ett dataspel är ett program som tagits fram främst i underhållningssyfte. I lärsituationer är det populärt att använda dataspel för att belysa vissa aspekter av lärande.

Dator

En dator är en maskin som utför beräkningar och bearbetar data enligt givna instruktioner. Den styrs av ett program och används numer till väldigt många olika tillämpningar. Datorer består primärt av elektroniska,

mekaniska och optiska delar och datorer kan variera mycket i storlek. Datorer finns numer överallt omkring oss i de flesta maskiner vi använder.

Exempel på datorer: persondator, miniräknare, mobiltelefon, lärplatta, spelkonsol.

Datorprogram

Se program.

Enkortsdator

En variant av datorer är små datorer för en specifik funktion där alla komponenter sitter på ett enda kretskort. Enkortsdatorer har ofta funktioner för att kommunicera med utrustning utanför datorn som t.ex. att styra motorer, lysdioder med mera och att kunna mäta saker som t.ex. temperatur eller ljusintensitet. Enkortsdatorer är användbara i skolan för att lära ut grundläggande programmering där eleverna får styra och mäta saker i den fysiska världen.

Exempel på enkortsdatorer: Arduino, Micro:bit, Raspberry Pi.

Funktion

En funktion, kallas även metod eller underprogram, är en del av ett program som kan anropas för att utföra en uppgift oberoende av resten av programmet. Används för att enklare kunna använda en isolerad del av programmet på flera ställen i programmet. En funktion kan returnera ett värde, t.ex. en beräkning eller bara utföra en uppgift. Användandet av funktioner i programmering är mycket vanligt och det är rekommenderat att strukturera upp programmet med hjälp av funktioner.

Grafiska element

I blockprogrammering används grafiska element, bilder, för att representera skriven kod.

Gränssnitt

Det sätt som en användare styr och interagerar med ett program kallas för gränssnitt. Alla program har någon form av gränssnitt, dessa kan vara grafiska med knappar och fönster, textbaserade, talbaserade eller haptiska, vilket betyder att de nyttjar vår känsel och kroppsrörelse.

Ett gränssnitt kan också finnas mellan olika delar i ett program eller mellan program.

Hårdvara

Hårdvara är de fysiska, hårda delarna som en dator består av.

Exempel på hårdvara: beräkningsenhet, hårddisk, minne, bildskärm, mus, tangentbord, pekplatta, skrivare.

Interpretator

En interpretator fungerar som en programtolk och översätter steg för steg de instruktioner som ett program beskriver till maskinkod som datorn förstår.

Exempel på programspråk som körs i en interpretator är Basic och JavaScript.

Iteration

Se repetition.

Kod och Koda

Kod är slang för själva programmet och koda är slang för att skriva själva programmet.

Kompilator

En kompilator är ett datorprogram som översätter från ett programspråk som människor kan skriva och läsa till maskinkod som datorn förstår.

Loop

Se repetition.

Maskinkod

Maskinkod, även kallat binärkod är det språk som en dator förstår och det består av ettor och nollor. Maskinkod är svårt för människor att läsa och förstå. Olika typer av datorer förstår olika typer av maskinkod.

Metod

Se funktion.

Mjukvara

Se program.

Mjukvaruutveckling

Se programmering.

Program

Program, kallas även programvara, mjukvara eller datorprogram och är en samling instruktioner som en dator kan tolka och utföra. Program kan vara mycket enkla med några få rader eller mycket komplexa med miljontals rader.

Exempel på program: ordbehandlare, bildredigeringsprogram, webbläsare, styrning av tvättmaskin, digital klocka.

Programmerare

En människa som utvecklar program. En skicklig programmerare förstår bland annat grunderna i programmering och kan enkelt lära sig nya programmeringsspråk.

Programmering

Programmering, även kallat mjukvaruutveckling är processen att utveckla ett program. Programmering är bredare än att bara skriva kod och innefattar förutom själva skrivandet att designa programmet, dvs. hur det ska delas upp i olika delar för att det ska bli lättare att förstå. Det innehåller också att analysera, dokumentera och förstå problemet som ska lösas.

För att programmet ska vara resurssnålt så är det viktigt att skriva effektiva lösningar som inte slösar datorns resurser vilket i sin tur betyder att t.ex. en mobiltelefons batteri tar slut snabbare.

Viktigast vid programmering är att förstå problemet och hitta en generell lösning som kan programmeras, dvs. arbeta med problemlösning.

Programmering utan dator

Programmering utan dator är en form av programmering som används för att lära ut principerna för programmering. Det kan till exempel handla om entydiga stegvisa instruktioner eller hur symboler kan representera en instruktion till exempel en ritad pil som betyder gå framåt.

Här används inte datorer utan istället används papper och penna eller tal för att förmedla instruktionerna. Instruktionerna utförs av människor som t.ex. flyttar sig själva eller en kompis eller flyttar en sak baserat på programmet.

Programmeringsmiljö

En programmeringsmiljö är en typ av program där en programmerare får hjälp att utveckla ett nytt program. Via programmeringsmiljöer är det oftast enkelt att testköra sitt program och se hur programmet som testkörs ser ut på olika typer av datorer.

Programmeringsmiljöer kan också vara grafiska där användaren arbetar med grafiska element för att bygga upp ett program.

Exempel på programmeringsmiljöer: Brackets, Eclipse, MonoDevelop, Unity, Visual Studio.

Exempel på grafiska programmeringsmiljöer: Mindstorms, Scratch, Snap!, Unreal.

Programspråk

Programspråk, kallas även programmeringsspråk, är språk som människor använder för att skapa program. Datorn förstår inte programspråk direkt utan detta måste översättas till maskinkod av en kompilator eller interpretator.

Olika programspråk är lämpade för olika tillämpningar och det finns idag mer än 750 olika programspråk.

Exempel på programspråk: Python, C, C++, Javascript, Java, Ruby och Blockly

Programvara

Se program.

Rekursion

Rekursion är ett programmeringsbegrepp som betyder att vi kör samma kod flera gånger, men med olika värden baserat på anrop till sig själv tills vi uppnår ett stoppvärde. Rekursion är vanligt vid implementation av algoritmer. Ett exempel är uträkning av matematisk fakultet som bygger på att vi multiplicerar tal i fallande storlek tills vi kommer ner till 1.

Repetition

Repetition kallas också iteration, slinga eller loop. En repetition är en programmeringsterm som betyder att vi upprepar en eller flera rader i programmet. Istället för att skriva samma del av programmet flera gånger så kan en repetition användas.

Slinga

Se repetition.

Sprajt eller sprite

En sprajt är en grafisk komponent som kan flyttas omkring på skärmen i ett program. Används främst i dataspel och mer modernt i blockprogrammeringsmiljöer.

Syntax

Syntax är en serie regler som beskriver hur ett program eller strukturerat dokument ska sättas samman för att anses vara korrekt.

Test

För att veta att ett program fungerar används olika tester som en del av utvecklingen av programmet. Testerna kan ske på separata delar av programmet eller på ett helt system bestående av flera olika program. Test inom programmering är viktigt för att minska antalet fel i programmet samt säkerställa dess funktion och göra det lättare att felsöka.

Textbaserad programmering

Textbaserad programmering är den vanligaste formen av programmering där programmeraren beskriver sitt program med hjälp av text (bokstäver, siffror och tecken).

Underprogram

Se funktion.

Utvecklingsmiljö

Se programmeringsmiljö.

Variabel

Variabel är en programmeringsterm och betyder en namngiven låda eller minne med ett värde som kan ändras i ett program. Lådan refereras till via sitt namn och är användbart om värdet ska användas på flera ställen i programmet. T.ex. poängräknare i ett spel eller hur många steg vi mätt upp i en stegräknare. Kan jämföras med en låda där vi kan lagra en sak i taget.

Vektor

Vektor, kallas också array, är en programmeringsterm. En vektor är en sekvens av ett godtyckligt antal värden som går att komma åt via ett ordningstal som anger en position i vektorn och där värdena oftast är av samma typ. Jämför med en variabel som bara kan innehålla ett enda värde.

Kan exemplifieras med en byrå med numrerade lådor där vi kan lagra saker i varje låda och enkelt läsa ut dessa med hjälp av numret på lådan.

Villkor

Villkor, som även kallas if-sats, är en programmeringsterm för att i ett program välja mellan olika alternativ. Det är ett logiskt uttryck för att avgöra om delar av programmet ska köras eller inte. T.ex. kontrollera om en spelares poäng är större än senaste högsta poäng i ett spel och i så fall, be spelaren skriva in sitt namn till topplistan. Ett annat exempel kan vara att kontrollera om användaren skrivit in rätt lösenord för att kunna fortsätta i programmet.