首頁

MySQL教學

網站技巧

網路程式設計

軟體程式設計

資料庫

作業系統

其它

Q

甘畑

首頁

科技

程式語言

# 高階語言-組合語言和機器語

## 言

其他 · 發表 2019-02-01

機器語言 <![endif]> (Machine language)

機器語言Machine Language(低階語 言)

機器語言是計算機唯 一能接受和執行的語 言。機器語言由二進 位制碼組成,每一串 二進位制碼叫做一條 指令。一條指令規定



统叶立座生



【卡片】ubuntu 安裝 steam "Missing 32-bit libraries: libGL.so.1"

[RK3288][Android6.0] 有線耳機驅動小結

Selenium3 簡明教程(Python版)

jquery類選擇器/基於下標的選擇器/層 次選擇器的使用場景

坑爹的MP3的crc校驗位

理解mysql資料庫和oracle資料庫幾個物件概念

Docker下redis的主從配置

Oracle 增加修改刪除欄位與添加註釋

了計算機執行的一個 動作。一臺計算機所 能懂得的指令的全 體,叫做這個計算機 的指令系統。不同型 號的計算機的指令系 統不同。

指令通常由幾個位元 組組成,第一個位元 組是操作碼,它規定 了計算機要執行的基 本操作;後面的位元 組是運算元,它規定 了操作物件或操作物 件的地址。

機器語言是直接用二 進位制程式碼指令表 達的計算機語言,指 今是用0和1組成的一 串程式碼,它們有一 定的位數, 並分成若 干段,各段的編碼表 示不同的含義,例如 某臺計算機字長為16 位,即有16個二進位 制陣列成一條指令或 其它資訊。16個0和1 可組成各種排列組 合,通過線路變成電 訊號,讓計算機執行 各種不同的操作。

如某種計算機的指令 為

10110110000000000,

Linux x86-64安裝oracle 11.2

聊聊高併發系統之限流特技

它表示讓計算機進行 一次加法操作;而指 令1011010100000000

則表示進行一次減法 操作。它們的前八位 表示操作碼,而後八 位表示地址碼。從上 面兩條指令可以看 出,它們只是在操作 碼中從左邊第0位算起 的第6和第7位不同。 這種機型可包含256 (=28)個不同的指 令。

使用機器語言編寫程式是一種相當煩瑣的工作,既難於記憶也難於操作,編寫出來的程式全是由0和1的數字組成,直觀性差、難以閱讀。不僅

難學、難記、難檢 查、又缺乏通用性, 給計算機的推廣使用 帶來很大的障礙。

最早的程式設計語言 都採用機器語言來編 寫。當時,要在計算 機上執行的所有演算 法都必須直接用機器 語言來表達,計算機 才能接受。演算法的 運算序列包括運算物 件和運算結果都必須 轉換為指令序列。其 中的每一條指令都以 編碼(指令碼和地址碼) 的形式出現。與演算 法語言表達的演算 法,相差十萬八千 里。對於沒受過程式 設計專門訓練的人來 說,一份程式恰似一 份"天書",讓人看了不 知所云,可讀性極 差。

用機器語言表達演算 法的運算、資料和控 制十分繁雜瑣碎,因 為機器語言所提供的 指令太初等、原始。 機器語言只接受算術 運算、按位邏輯運算 和數的大小比較運算 和數的大小比較運算 等。對於稍複雜的運 算,都必須一一分

解,直到到達最初等 的運算才能用相應的 指令替代之。機器語 言能直接表達的資料 只有最原始的位、位 元組、和字三種。演 算法中即使是最簡單 的資料如布林值、字 元、整數、和實數, 也必須一一地對映到 位、位元組和字中, 還得——分配它們的 儲存單元。對於演算 法中有結構的資料的 表達則要麻煩得多。 機器語言所提供的控 制轉移指令也只有無 條件轉移、條件轉 移、進入子程式和從 子程式返回等最基本 的幾種。用它們來構 造迴圈、形成分支、 呼叫承式和過程得事 先做許多的準備,還 得靠許多的技巧。

直接用機器語言表達演算法有許多缺點。

Ø 大量繁雜瑣碎 的細節牽制著程式設 計師,使他們不可能 有更多的時間和精力 去從事創造性的勞 動,執行對他們來說 更為重要的任務。如 確保程式的正確性、 高效性。

- Ø 程式設計師既 要駕馭程式設計的全 域性又要深入每一個 區域性直到實現的細 節,即使智力超群的 程式設計師也常常會 顧此失彼,屢出差 錯,因而所編出的程 式可靠性差,且開發 週期長。
- Ø 由於用機器語言進行程式設計的思維和表達方式與人們的習慣大相徑庭,只有經過較長時間職業訓練的程式設計師才能勝任,使得程式設計曲高和寡。
- Ø 因為它的書面 形式全是"密"碼,所以 可讀性差,不便於交 流與合作。
- Ø 因為它嚴重地 依賴於具體的計算 機,所以可移植性 差,重用性差。

這些弊端造成當時的 計算機應用未能迅速 得到推廣。 組合語言(Assember language) 組合語言Assembler Language(低階語 言)

為了克服機器語言上 述的缺點,出路在於 程式設計語言的抽 象,讓它儘可能地接 **近於演算法語言。為** 此,人們首先注意到 的是可讀性和可移植 性,因為它們相對地 容易通過抽象而得到 改善。於是,很快就 出現組合語言。這種 語言對機器語言的抽 象,首先表現在將機 器語言的每一條指令 符號化:指令碼代之以 記憶符號, 地址碼代 之以符號地址,使得 其含義顯現在符號上 而不再隱藏在編碼 中,可讓人望"文"生 義。其次表現在這種 語言擺脫了具體計算 機的限制,可在不同 指令集的計算機上執 行,只要該計算機配 上組合語言的一個彙 編程式。這無疑是機 器語言朝演算法語言 靠攏邁出的一步。但 是,它離演算法語言

還太遠,以致程式設計師還不能從分解演算法的資料、運算和控制到彙編才能直接表達的指令等繁雜瑣碎的事務中解脫出來。

例如,要計算**c=7+8**,可以用如下幾條彙編命令:

標號 指

令 說明

START GET

**7**; 把**7**送進累 加器**ACC**中

ADD 8;

累加器ACC+8送進累 加器ACC中

PUT C;

把累加器ACC送進C中

**END** 

STOP; 停機

其中(ACC)表示累加器中的值,等等。

高階語言(High-level language) 高階語言

機器語言和組合語言 都是面向機器的,高 階語言是面向使用者 的。到了50年代中 期,出現程式設計的 高階語言如Fortran, Algol60,以及後來的 PL/I,Pascal等,演算 法的程式表達才產生 一次大的飛躍。用高 階語言源程式可 問語言源程式可 必須翻譯成機器語言 目標程式才能被計算 機執行。高階語言流 機執行。高階語言 機執行兩種方式:編 譯方式和解釋方式。

- ∅ 編譯方式:先 由編譯程式把高階語 言源程式翻譯成目標 程式,執行時執行目 標程式。
- Ø 2.解釋方式: 在執行高階語言源程 式時,由解釋程式對 源程式邊翻譯邊執 行。

轉成機器語言。組合 語言作為一種中介語 言, 並沒有獲得很大 成功,原因是它離演 算法語言還太遠。這 便指引人們去設計一 種儘量接近演算法語 言的規範語言,即所 謂的高階語言,讓程 式設計師可以用它方 便地表達演算法,然 後藉助於規範的高階 語言到規範的機器語 言的"翻譯",最終將演 算法表達為機器語 言。而且,由於高階 語言和機器語言都具 有規範性,這裡的"翻 譯"完全可以機械化地 由計算機來完成,就 像組合語言被翻譯成 機器語言一樣,只要 計算機配上一個編譯 程式。

步要做的只是將第一步得到的高階語言程式翻譯成機器語言程式。至於程式設計師如何用高階語言表達如何用高階語言表達如何將高階語言表達如何將高階語言表達如的演算法翻譯成機器語言表達的演算法,顯然毫不相干。

為了解決低階語言的 缺點,人們為了從根 本上擺脫語言對機器 的依賴,使之獨立於 機器,由面向機器改 為面向過程,經過多 年精心的研究,終於 在1954年首先創造出 一種與具體的計算機 指令系統無關的、表 達方式或接近於人們 對求解過程或問題的 描述方式,且易於掌 握和書寫的語言,這 就是FORTRAN語言。 人們把具有以上特點 的語言稱為高階語 言,這就是計算機語 言系統中的第三代語 言。

例如計算A=1+2,若用高階語言(如BASIC語言)編寫,只要兩條語句:

10 A=1+2

#### 11 END

就可以完成A=1+2的 操作。這種語言,人 們易於掌握和理解, 即使小學生也容易掌 握,便於推廣,它不 再是面向機器,而是 "面向過程",即使用者 不僅要告訴計算機"做 什麼",還要告訴計算 機"怎麽做",也就是把 每一步操作事先設計 好,然後再編寫程 式,讓計算機按照指 定的步驟去執行。目 前,世界上已經有幾 百種不同型別、功能 各異的高階語言。使 用較多的有:

**BASIC** 

**FORTRAN** 

**PASCAL** 

C

PL/I

**ADA** 

LISP

程式設計語言從機器語言到高階語言的抽象,帶來的主要好處是:

高階語言接近演算法 語言,易學、易掌 握,一般工程技術人 員只要幾周時間的培 訓就可以勝任程式設 計師的工作; 高階語言為程式設計 師提供了結構化程式 設計的環境和工具, 使得設計出來的程式 可讀性好,可維護性 強,可靠性高; 高階語言遠離機器語 言,與具體的計算機 硬體關係不大,因而 所寫出來的程式可移 植性好,重用率高; 由於把繁雜瑣碎的事 務交給了編譯程式去 做,所以自動化程度 高,開發週期短,目 程式設計師得到解 脫,可以集中時間和 精力去從事對於他們 來說更為重要的創造 性勞動,以提高程式 的質量。

程式語言 ☑的種類之一,由被稱作翻譯器 (interpreter)的軟體 ☑

一邊把人類記述的程 式變換成電腦 ☑ 可執 行的形式,一邊執行 的形式的語言。

解釋語言是電腦 ②使用的機器語言 1對1地對應,與機器語言 2相比對應的 1對種的 1對種類 1 地對應,與人類語言 2相比與人類語言 1 對應,與人類語言 1 對應,與人類語言,與人類語言,與人類語言,則人類所以由解釋語,所以由解釋語,所以由解釋語,則以由解釋語,則以由解釋語,則以由解釋語,則以由解釋語,則以由解釋語,則以由解釋語,則以由於實際 2 對於數學,可執行的程式。

高階語言裡還有,一 併地轉換成機器語言 的程式之後再執行的 編譯語言 🗹。這個語 言,由於需要先把程 式轉換成機器語言程 式然後再執行,所以 在開發的速度與容易 度,修正的方便性上 都不如組合語言 🗹, 但是,由於不需要同 時翻譯,所以在實效 速度上佔有很大的優 勢。還有,解釋語言 中的執行環境区裡需 要原始碼,編譯語言 ☑裡就不需要。

#### 標籤:

### ௴ 您可能也會喜歡...

【如何讓程式碼 Haskell心得

變"高階"(一)】- (1)模式匹

Spring組合註解 配、高階函式、

提升程式碼維度 組合函式

SpringBoot5- three.js使用掩碼

spring高階話題- MaskPass高階效

組合註解與元註 果組合器

解,@Enable\*註 組合數——51nod

解的工作原理, 1120 機器人走方

測試 格 V3 (卡特蘭

數)

高階語言.組合語 高階語言,組合

言和機器語言 語言,機器語言

學程式設計的起 程式碼生成器技

點——高階語言 術亂彈十三,郎

大鍋燴 騎竹馬來,高階

語言和超級語言

SQL高階語言查 深入理解計算機

組合語言和機器

語言

® 漢語是世界上 JAVA高階語言隨

唯一一種面向物 著JDK升級新增

件的高階語言 特性

【FIF\_Bro的部落 從彙編和高階語

格】C++ 的最大 言的角度理解傳

優點是,它是一 值方式,傳值,

門強大,兼顧了 傳引用,傳指標

底層效率的,高 的本質機制與區

階語言。這使得 別。白話通俗易

C++ 幾乎沒有任 懂。

何侷限性,無所

不能。只要你是

一個追求極致達

到偏執的人,你

幾乎總可以追尋

到它的影子和存

在。這門語言超

高的開發成本,

使得只有偏執的

人,才會對它痴

狂

首頁

Python教學



ITREAD01.COM © 2018. 版權所有。