

Storia e fondamenti dell'informatica.

*Corso di formazione per tecnico consulente informatico per sistemi ERP
open source.*

Giovanni Marelli

Padova November 7, 2013

Struttura

- Storia
- Utilizzi
- Sistemi
- Linguaggi
- Programmazione

Storia

Evoluzione dei calcolatori

- Macchina analitica (Charles Babbage, 1791-1871)
- Progetto Manhattan (Los Alamos, 1939-1945)
- Elea 9003 (Olivetti, 1957)
- Unix (1964)
- Darpanet (commutazione di pacchetto, 1969)
- PC (IBM libero, 1984)
- X (MIT, 1984)
- Macintosh/Mac OS (1984)
- Desktop (1984)
- GNU/GPL (1989)
- Web/http (CERN, 1991)

Utilizzi

Principali utilizzi del calcolo

- Calcolo numerico
- Gestione dati
- Analisi dati
- Simulazioni
- Ottimizzazione
- Web

Calcolo numerico [c, c++, CGAL, gsl, matlab]

- Struttura (statica, equazioni differenziali alle derivate parziali)
- Aerodinamica (Navier-Stokes)
- Deformazioni plastiche (elementi finiti)
- Calcolo simbolico (mathematica, maple)

Banche dati [SQL]

- Utenti
- Afflusso (code, accessi)
- Transizioni
- Rilevazioni (meteo, utilizzo)

Analisi

- Medie
- Varianze
- Distribuzioni empiriche
- Inferenza statistica
- Correlazioni

Simulazioni (in parallelo su cluster di calcolo multiprocessore [c, c++, fortran])

- Meteorologia, cosmologia
- Sistemi complessi
- Semiconduttori, materia soffice
- Strutture molecolari
- Previsioni (affluenze, code)

Modellizzazione, distribuzioni teoriche

Ottimizzazione (Monte Carlo, minimizzazione [c, c++])

- Disegno delle linee e dei percorsi (autobus, trasporti, corrente)
- Calcolo degli orari
- Distribuzione della banda
- Utilizzo delle centrali elettriche

Reti

- Scambio dati
- Servizi posta, chat

Reti locali

- Gateway (192.168.0.1)
- Broadcast
- Alias
- Condivisione (stampanti, dischi, ...)
- Terminale

Web

- dns (distinzione tra indirizzi locali e web)
- Provider
- Server

Sistemi

Sistemi operativi

- UNIX (1969)
- GNU/Linux (1991)
- Mac, iOS
- Windows
- BeOS/Haiku
- Android
- firmware
- PLC

Struttura

- bios
- kernel
- moduli
- programmi
- GUI interfaccia grafica

Funzionalità

- Minima memoria fissa (EPROM) o scrivibile
- Carica le caratteristiche minime
- Carica il bootloader

Selezione al boot

- Monta le periferiche avviabili
- Carica le immagini di boot dei sistemi operativi
- Selezione dei sistemi operativi disponibili
- Esegue il kernel

fornisce ai processi un accesso all'hardware

- scheduling, multitasking
- HAL (hardware abstraction layer) diversi livelli per l'accesso all'hardware

Creato per studiare la dinamica del sole e dei pianeti e di piccoli corpi (Bell lab 1969).

Caratteristiche

- Multiuser
- Multitasking
- Portabile
- Basato su c

Storia

- 3.x (1990)
- Portabilità

Storia

- kernel linux (Linus Torvalds)
- sistema operativo GNU (Richard Stallman, 1984)
- licenza GPL (e derivati)
- scritto principalmente in c
- la shell è lo strumento base

Strumento base, conchiglia attorno al kernel

- Ogni programma può essere eseguito da linea di testo
- Uso degli script, automazione
- autocompletamento, history
- multiutente, multitasking, locale, remoto
- file di configurazione facili da scambiare

Permesso d'autore

- GPL (1989)
- Copyleft (Richard Stallman, 1989)
- Creative commons (2001)

Tutele

- Riconoscimento dell'autore
- Condivisione
- Estensione con la stessa licenza

Opzionali

- Non commerciabilità
- Non derivabilità

Gestore grafico

- X11, XFree, X.org
- Client/server
- Non prevede l'interfaccia utente
- Visualizzazione e posizionamento è lasciato al window manager (GNOME, KDE, xfce)

Linguaggi

Compilati, basso livello

- assembly
- c (1969), c++
- fortran (1977)

Run time, alto livello

- python
- perl
- java
- javascript

Script

- matlab, mathematica, maple

Lato server

- php
- MySQL
- python

Lato client

- javascript
- perl
- HTML

Programmazione

Stili di programmazione

- Imperativo, monolitico (assembly)
- Funzionale (Lisp 1958, php)
- Strutturato, modulare (c, fortran)
- Ad oggetti (c++, java, python)

L'algoritmo è una sequenza di istruzioni per compiere un calcolo

- Ada Lovelace, Charles Babbage, Luigi Menabrea sulla macchina analitica (1843)
- Macchina di Turing (1936), una sequenza di istruzioni I/O e un processore come strumento ideale per la definizione di un algoritmo

```
double d(double x,int N){
  double d = 0.,fact1 = 1.,fact2 = 1.;
  for(int i=0;i<N;i++){
    fact1 *= (x - 1.);
    fact2 *= x;
    d += fact1/fact2;
  }
  return d;
}
```

$$d_N(x) = \sum_{i=0}^N \frac{(x-1)^N}{x^N}$$

Concetti di programmazione

- Memoria
- Variabili
- Compilazione
- File oggetto
- Librerie
- Linking
- Debugger
- Profilazione

Cache

- Memoria messa a disposizione del processore in run time
- Il livello di cache determina la velocità di accesso alla memoria

Fissa

- Legge e scrive su disco

RAM

- Memoria condivisa dal sistema

Multiprocessore

- Evitare gli scambi di informazioni tra memorie di processori diversi (lento)

Variabile

- Posizione di memoria: fissa o dinamica allocata in compilazione o in run time
- Tipo: modalità in cui viene interpretata la sequenza di bit (segno, esponente, mantissa, ...) (int 1B, float 1B, double 2B, bool 1b, char 1B, ...)

Locali/globali

- Locali: vengono lette e scritte solo all'interno della funzione/classe
- Globali: occupano uno spazio fisso nel programma e possono essere lette/scritte da qualsiasi funzione o classe (pericoloso!!)

Puntatori

- Occupano uno spazio in memoria e un indirizzo che può essere associato ad un'altra locazione di memoria
- Serve per leggere e scrivere altre variabili dello stesso tipo

Importantissimo: imparare a dare nomi sensati alle variabili!!!!

Dichiarazione

```
double sqr(double a){  
    return a*a;  
}
```

- Riceve degli argomenti
- Definita da un tipo
- Ritorna lo stesso tipo
- Si instancia

Oggetto

- Ha le proprie variabili, funzioni, costruttore, distruttore
- L'oggetto va pensato come struttura a se stante, da utilizzare come scatole chiuse
- Ereditarietà

```
class Telefono:Comunicazioni{  
    public:  
    Telefono();  
    ~Telefono();  
    bool SeLinea,SeAcceso;  
    float Mic;  
    float Audio;  
    int Numero;  
};
```

Definizione più generale del problema, più delicato

```
(defun factorial (n)
  (if (<= n 1)
      1
      (* n (factorial (- n 1)))))
```

```
(defun pow (x n)
  (if (<= n 1)
      x
      (* x (pow (x) (- n 1))))
  )
)
```

Prima della compilazione

- Il preprocessore sostituisce esplicitamente la macro con la stringa a cui si riferisce
- Ricordano le funzioni inline [c++], static [c]

Opzioni per il compilatore

- static
- restrict
- virtual

Opzioni di compilazione

- -w warnings
- -o0 -o1 -o2 ottimizzazione
- -g debugger

Bug

- Segmentation fault: lettura oltre gli elementi di memoria allocati
- Arithmetic exception: incongruenze come $1./0.$ o $\text{sqrt}(-1.)$ (possono essere disabilitati in compilazione)

gdb

- bt backtrace
- breakpoint

Output del programma su shell o su file

- Ottimo strumento per gestire gli errori, controllare gli accessi
- `/var/log/` per i principali programmi

Nella scrittura:

- Evitare il calcolo di \sin \cos $\sqrt{\quad}$ e $1./$ (lenti)
- Trovare la migliore struttura per allocare i dati per evitare i salti di memoria

Sviluppo di GUI

- Qt, GTK+, Motif (1980)
- libpng, libtiff, libcairo

3D

- openGL
- compiz

Librerie scientifiche

- gsl (matematica)
- CGAL (trasformazioni e ordinamento geometrico)
- Elmer (elementi finiti)
- splines, bezier

API/Application programming interface

- Librerie standard

Commentare

- Scrivere nel sorgente la licenza, il contenuto del file, lo scopo delle funzioni e passaggi delicati

Documentare

- doxygen (documentazione automatica, pdf, html)
- wiki (editata dagli utenti)
- blog

Evoluzione di un programma

- fork (costola di un programma)
- trunk (versione più sviluppata)
- branch
- stable (versione +/- fuori dallo sviluppo)

Sviluppo di gruppo (subversion, bazaar)

- commit (pubblica un file)
- update (aggiorna la macchina locale con i cambiamenti)
- diff (mostra la differenza tra le versioni)
- add/del (aggiunge/rimuove un file dal repository)

Reti

Struttura

- Server (nodi)
- Provider
- Pacchetto
- Client (host)
- DNS (domaine name service) NAT (network address translation, modifica gli IP dei pacchetti)

Contiene:

- Database: contenuti relativi ad utenti, materiale
- Programmi lato server: php, python

Servizi/porte/ protocolli TCP/IP

- mail uscita (smtp)/entrata (pop, imap)
- http
- ssh
- ftp (file transfer protocol)
- dav

Accesso:

- Porte/protocolli: pop, imap, smtp, ssh, stl, http, https
- Programmi lato client: browser, ssh

Il client accede al server e utilizza i programmi remoti

Il browser:

- Esegue il codice html, xml, javascript
- php viene interpretato sul server

Criptazione

- SSL/TLS (Secure Sockets Layer, Transport Layer Security)
- Chiave pubblica/privata gpg
- Firewall (controlla l'accesso ad ogni porta)

Malware

- Virus (riscrive il suo codice su file eseguibili ogni volta che un file infetto viene aperto)
- Worm (si diffonde tramite mail e internet e rimane attivo come processo di sistema)
- Trojan (programma sull'host con permessi eseguito da remoto)
- Dialer (si connettono a provider a pagamento)
- Spyware (forniscono informazioni sull'utente, a volte parte di un freeware)

Linux Apache MySQL php

- Sistema operativo
- Web server
- Database server
- Programma lato server

Web server

- Gestisce le richieste dai client (traduce, controlla, contenuto (MIME type), risponde)
- httpd.conf (moduli, estensioni, mime-type)
- .htaccess (autorizzazione, riscrittura degli url, cache)
- proxy (gestisce le autorizzazioni e reindirizzamenti)
- socket su IP per la lettura e scrittura di dati per utilizzare API standard

MySQL, PostgreSQL

- Gestisce gli utenti e gli accessi
- Performante in termini di memoria e ricerca
- libero

Oracle

- Sistema di gestione base dati
- *tablespace* contiene *datafile* con *segment* di *extent* (gruppi contigui di *data block*)