

Appunti di Fondamenti di Informatica

Giacomo Simonetto

Primo semestre 2023-24

Sommario

Appunti del corso di Fondamenti di Informatica della facoltà di Ingegneria Informatica dell'Università di Padova.

Indice

1	Storia dell'informatica	3
2	Algoritmo	3
3	Computational Thinking	3
4	Computer	4
5	Modello di Von Neumann	4
5.1	Central Processing Unit o CPU	4
5.2	Memoria primaria e secondaria	5
5.3	Dispositivi di I/O	6
6	Sistemi operativi	7
6.1	Unix e Linux	7
7	Rappresentazione delle informazioni nei calcolatori	7
7.1	Sistema posizionale	7
7.2	Rappresentazione in modulo - segno	7
7.3	Rappresentazione in complemento a 2	7
7.4	Rappresentazione in virgola fissa	7
7.5	Rappresentazione in virgola mobile	7
8	Linguaggi di programmazione	7
9	Java	7

1 Storia dell'informatica

I primi tentativi della ricerca di un linguaggio formale risalgono alla fine del 1800. Ci si chiede se la matematica sia un sistema formale completo e se esiste un procedimento meccanico (passo-passo, finito) per dimostrare se una proposizione sia vera o falsa.

Il primo tentativo di "*formalizzazione della matematica*" viene svolto da David Hilbert, con cui si scopre che la matematica possiede 23 problemi di formalizzazione chiamati "*23 problemi di Hilbert*".

La risposta alla prima domanda risale al 1931 quando Goedel, con il "*teorema di incompletezza*" conferma che la matematica non è un sistema formale.

Nel 1936 Church, Turing e Kleene elaborano dei formalismi meccanici tra cui la *Macchina di Turing* e la *Tesi di Church-Turing* che sostiene che tutto ciò che è computabile è computabile dalla macchina di Turing universale.

Nel 1943 si arriva a costruire l'*ENIAC*, il primo computer (general purpose) della storia.

La capacità computazionale tra una macchina di Turing e un computer odierno è la stessa (eccetto per il fatto che la macchina di Turing prevedeva uno spazio di archiviazione illimitato), cambia solo la velocità computazionale. Entrambe le macchine risolvono gli stessi problemi, ovvero tutti quelli che si possono risolvere con un algoritmo.

2 Algoritmo

Un algoritmo è un metodo di risoluzione di un problema che:

- deve essere eseguibile
- non deve essere ambiguo
- deve concludersi in un numero finito di passi

3 Computational Thinking

Per computational thinking, o pensiero computazionale, si intende l'insieme delle abilità che permettono di astrarre il problema e tradurlo in algoritmo. Comprende le tecniche di astrazione/risoluzione di problemi algoritmici tra cui la decomposizione di problemi complessi e la modularità.

4 Computer

Per computer, o calcolatore, si intende un sistema di elaborazione e memorizzazione di informazioni che opera sotto il controllo di un programma. È composto da hardware (parte fisica) e software (programmi e dati). I dati possono essere di diverso tipo (immagini, testi, audio, video, ...) e sono rappresentati elettricamente in 0 e 1.

Esistono diversi tipi di computer (workstation, smartphone, ...) che possono svolgere diversi tipi di impieghi (elettrodomestici, giochi, fotografie, ...).

5 Modello di Von Neumann

Il modello di Von Neumann è una rappresentazione dell'architettura di un elaboratore. Prevede la presenza di 4 blocchi: la CPU, la memoria primaria, la memoria secondaria e i dispositivi di I/O collegati insieme grazie al BUS.

Inoltre sono presenti due diversi flussi di informazioni: quello di dati è bidirezionale, mentre quello degli indirizzi e dei segnali di controllo è unidirezionale con direzione CPU → altri dispositivi.

5.1 Central Processing Unit o CPU

Compiti

La Central Processing Unit ha il compito di:

- individuare ed eseguire le istruzioni
- elaborare dati attraverso la ALU (Unità Logico Aritmetica)
- reperire dati di input e restituire dati di output

Blocchi

È costituita da tre blocchi:

ALU:	anche chiamata <i>Arithmetic Logical Unit</i> , risolve le espressioni logico - algebriche come somma, differenza, prodotto, divisione, and logico, or logico, ...
<hr/>	
	memoria temporanea per dati che devono essere subito elaborati:
	- il Memory Address Register , o <i>MAR</i> , che memorizza l'indirizzo dell'istruzione da eseguire o del dato da utilizzare
Registri:	- il Memory Data Register , o <i>MDR</i> , che memorizza i dati/le istruzioni lette dalla memoria primaria
	- l' Accumulator , o <i>ACC</i> , che memorizza dati temporaneamente per essere utilizzati nelle istruzioni successive
<hr/>	
	anche detta unità di controllo, è unità principale della CPU, è costituita da:
Control Unit:	- Program Counter , o <i>PC</i> , che memorizza l'indirizzo dell'istruzione successiva da eseguire
	- Instruction Register , o <i>IR</i> , ovvero il registro che memorizza l'istruzione in esecuzione

Funzionamento

La CPU ha funzionamento ciclico che si divide in tre fasi. La velocità di una CPU, chiamata frequenza di clock è espressa in cicli al secondo (dell'ordine dei GHz) ed è scandita dal *Clock*. La velocità massima è dovuta ai limiti fisici della tecnologia disponibile.

1° fase	fetch	<ul style="list-style-type: none">- viene letto l'indirizzo dell'istruzione da eseguire dal <i>PC</i> e viene salvato nel <i>MAR</i>- viene incrementato il <i>PC</i> in modo che punti all'istruzione successiva- viene letta e caricata l'istruzione prima nel <i>MDR</i> poi nell'<i>IR</i>
2° fase	decode	<ul style="list-style-type: none">- la <i>CU</i> decodifica l'istruzione salvata nell'<i>IR</i>- se necessario viene caricato nel <i>MAR</i> l'indirizzo del dato da elaborare o della posizione in cui scrivere il dato elaborato
3° fase	execute	<ul style="list-style-type: none">- viene eseguita l'istruzione:- se necessario viene caricato nel <i>MDR</i> il dato referenziato dal <i>MAR</i>- il dato può essere salvato nell'<i>ACC</i> o impiegato in un'operazione logico-algebrica eseguita dalla <i>ALU</i>- il risultato viene salvato nell'<i>ACC</i>- oppure il dato memorizzato nell'<i>ACC</i> viene scritto nell'indirizzo di memoria contenuto nel <i>MAR</i>

Limiti e parallelismo

I limiti della CPU sono principalmente due: la frequenza di clock e l'impossibilità di eseguire un'istruzione, finché non viene completata quella precedente. Per superare il secondo problema si sono cercate soluzioni come il parallelismo. Il parallelismo si divide in:

- **parallelismo a livello di istruzioni:**

detto anche pipeline o multiscalari, consiste nel suddividere il ciclo di un processore in 5 stadi (lettura, decodifica, recupero operandi, caricamento, esecuzione, invio risultati) e di eseguire contemporaneamente più istruzioni su stadi diversi. In questo modo non è necessario aspettare che l'esecuzione dell'istruzione precedente termini per iniziare quella della successiva, ma è sufficiente che sia completato il primo stadio.

- **parallelismo a livello di processori:**

consiste nell'avere più processori che lavorano contemporaneamente in grado di eseguire più istruzioni nello stesso momento. In base all'architettura di distinguono in multiprocessori (se sono presenti più processori che condividono la stessa memoria) o multicomputer (se sono più processori, ciascuno con la propria memoria dedicata, collegati tra loro).

5.2 Memoria primaria e secondaria

Compiti

La memoria ha il compito di memorizzare dati e programmi, sia in maniera temporanea, che permanente.

Struttura

La memoria è composta da celle chiamate allocazioni di memoria. Ogni allocazione può contenere un preciso numero di bit. Un bit (abbreviazione di Binary Digit) è l'unità minima di dimensione della memoria e corrisponde allo spazio occupato da 0 o 1. Il bit è un sottomultiplo del byte, 1byte = 8bit. Il byte è l'unità minima di accesso singolo alla memoria ed è l'unità base per la misura della dimensione dello spazio di archiviazione. I suoi multipli sono kilobyte (KB), megabyte (MB), gigabyte (GB), terabyte (TB), ...

Memoria primaria

La memoria primaria è la più veloce delle due, ma anche la più costosa. Ne esistono due tipi:

RAM	o <i>Random Access Memory</i> , memoria volatile, dotata della caratteristica di avere un tempo di accesso ad una cella indipendente dal luogo in cui essa si trova (tempo di accesso "casuale"). Viene impiegata per salvare dati temporanei derivati dall'esecuzione di programmi.
ROM	o <i>Read Only Memory</i> , memoria permanente di sola lettura in cui vengono salvati i programmi necessari all'avvio della macchina, es. BIOS (<i>Basic Input Output System</i>)
Cache	o <i>memoria di località</i> memoria estremamente veloce che permette di memorizzare celle di memoria che potenzialmente potrebbero tornare utili nelle future elaborazioni. Esistono due tipi di località: <ul style="list-style-type: none">- località temporale: accedere alla stessa cella in tempi vicini- località spaziale: accedere a celle limitrofe

Memoria secondaria

Memoria non volatile, più lenta e molto meno costosa della memoria primaria. È riservata all'archiviazione di file, dati, programmi (tra cui anche il sistema operativo) che vengono trasferiti nella *RAM* al momento dell'esecuzione. Esistono diversi supporti di archiviazione di memoria secondaria:

- *HDD* o disco magnetico
- *SSD* o disco a stato solido (solid state drive)
- dischi ottici come *CD*, *DVD*, *Blue-Ray*
- chiavette USB
- nastri magnetici, impiegati per l'archiviazione di documenti, sono molto lenti, ma hanno costo molto basso ed elevata capacità di archiviazione

Gerarchie di memoria

Maggiore è la dimensione, minore è la velocità ed il costo. In ordine crescente per velocità e costo e decrescente per dimensione:

- nastri magnetici
- dischi magnetici, ottici
- memoria centrale (*RAM*)
- cache
- registri del processore

5.3 Dispositivi di I/O

Permettono l'interazione dell'essere umano con la macchina. Comprendono mouse, tastiera, touchpad, schermo, stampante, ... Le operazioni relative ai dispositivi I/O sono:

polling	o controllo da programma, consiste nel ripetuto e periodico controllo dello stato dei dispositivi
interrupt	richiama l'attenzione della CPU attraverso un'interruzione del flusso di esecuzione
DMA	o <i>Direct Memory Access</i> , dispositivo indipendente dalla CPU che gestisce il flusso di dati dei dispositivi di I/O ed alleggerisce il carico della CPU (la CPU indica solo indirizzi e dati da spostare)

6 Sistemi operativi

6.1 Unix e Linux

...

7 Rappresentazione delle informazioni nei calcolatori

7.1 Sistema posizionale

...

7.2 Rappresentazione in modulo - segno

...

7.3 Rappresentazione in complemento a 2

...

7.4 Rappresentazione in virgola fissa

...

7.5 Rappresentazione in virgola mobile

...

8 Linguaggi di programmazione

...

9 Java

...