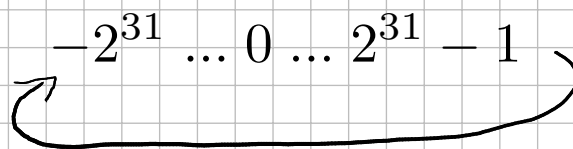


https://www.cs.unibo.it/~renzo/so/materiale2324/domande/20230928_domande_unix_c

RAPPRESENTAZIONE DEI DATI

Come vengono rappresentati in memoria gli interi con e senza segno?



I negativi si salvano in complemento a 2

0x12345678



BIG endian

LITTLE endian

Big endian / little endian, cosa sono?

fino a quando non succede che...
non dovete far collaborare due
macchine diverse...
(un esempio lo vediamo al corso di Reti)

A cosa servono i numeri ottali o esadecimali?

Se non ci fossero dovremmo scriverli solo in BINARIO :D ?

E quanti errori faremmo (sia il prof sia noi)? risposta TANTI, TANTISSIMI!

Le operazioni & (and), | (or), ~ (not): come funzionano?

Intanto dobbiamo distinguere tra operazioni ARITMETICHE e LOGICHE

& (and), | (or), ~ (not) sono operazioni ARITMETICHE.

A & B	0x34 & 0xC5
0x34	00110100
0xC5	11000101
AND	00000100

A B	0x34 0xC5
0x34	00110100
0xC5	11000101
OR	11110101

A	~A (NOT A)
0	1
1	0

"Gli informatici sono uomini che sussurrano ai processori" cit. Renzo Davoli :3

FLAGS

- è acceso?
- accendilo / spegnilo
- sono tutti accesi?
- ce n'è almeno uno acceso?

queste cose (giocando con i FLAGS)
si fanno comodamente con questi operatori

prendiamo un flag di 8 byte dove:



Vogliamo scrivere:

'se ho fame vado a magnà'

ho fame

ho sete

ho sonno

```
if (flag & 0x08)
```

```
    mangia()
```

00001000

[ovvero controlla questo bit (quello a 1)]

per questo parliamo di MASCHERE e non intendo quelle del teatro o le mascherine

le maschere in tipografia --> fogli con buchi x vedere determinate cose

falso è 0, vero è qualsiasi cosa diversa da 0.

2

```
if (flag & 0x80) 'oh cavolo' prima avevo scritto 0x80 e nessuno mi aveva  
    mangia()      corretto... ma come si fanno le cose CORRETTE?
```

scrivereif (flag & HO_FAME) mangia()così è più leggibile :3

--> questo simbolo io (Renzo Davoli) lo chiamo SHARP!

quindi mi sentirete dire /sharp define/ invece che /... define/

```
# define HO_FAME 1 << 3
```

ovvero definisco HO_FAME come la costante che ha

1 spostato a sinistra di 3 posizioni

Come si fa ad "accendere" bit?flag |= HO_FAME

dopo che ho mangiato [mangia()] vorrò spegnerlo...

Come si fa a "spegner" bit?flag &= ~ HO_FAME

e se voglio dire: 'se HO_FAME or HO_SETE allora mangia()' cosa scrivo?

Come si fa a considerare solo alcuni bit di un intero (mascheramento)?

```
#define OK_PASTO (HO_FAME | HO_SETE)
```

```
if ( (flag & OK_PASTO)==OK_PASTO)        questo è MOLTO USATO
```

--> per approfondire cercate 'come spaccare i bit in 4' sulla wiki del corso

<https://www.cs.unibo.it/~renzo/so/misc/bit4.pdf>

(come ho trovato il link?)

sito di Renzo Davoli

<https://www.cs.unibo.it/~renzo/>

corso di Sistaemi Operativi

<https://www.cs.unibo.it/~renzo/so/>

wiki del corso

[so.v2.cs.unibo.it --> https://so.v2.cs.unibo.it/wiki/index.php/Main_Page](https://so.v2.cs.unibo.it/wiki/index.php/Main_Page)

'Esempi, Esercizi ed Esperimenti 2023/24'

https://so.v2.cs.unibo.it/wiki/index.php/Esempi,_Esercizi_ed_Esperimenti_2023/24

'Esempi sulla rappresentazione dei dati'

https://so.v2.cs.unibo.it/wiki/index.php/Esempi_sulla_rappresentazione_dei_dati

'Come spaccare i bit in 4'

<https://www.cs.unibo.it/~renzo/so/misc/bit4.pdf>

Come fa l'operazione ^ (xor)?

v ^= 0xA1010 (A in binario)

flippa questi bits (quelli a 1)

in inglese si dice TOGGLE

Come si può usare il mascheramento per fare l'operazione di modulo di una potenza di 2?

3

Il MASCHERAMENTO può servire per fare l'operazione MODULO

$v \% 256$

questa è l'operazione di MODULO e farla è pesante...

(compilatore furbo se fa l'altro metodo)

$v \& 255$

ovvero 'guarda solo gli ultimi 8 bit'

si fa in un ciclo di macchina

quindi...

dati A e B come si fa a controllare se alcuni bit di A sono accesi in B?

`if (B & A)...`

dati A e B come si fa a controllare se tutti i bit di A sono accesi in B?

`if (B & A) == A)...`

GIT --> mandato volantino sul gruppo telegram di quest'anno (2324)



link al gruppo telegram:

<https://t.me/+HRseokr7sxFhOWUy>

link al messaggio:

<https://t.me/c/1933494044/103>

inviato da @LLibera (t.me/llibera)

messaggio di descrizione:

🔗 link typeform per iscrizione
(<https://lr533gb3hpt.typeform.com/to/m5fzNCsW>)

@admstaff_Chat per info/domande

(di cui Samu ha fatto reply per specificare e dare spoiler su 'TERMINALE AVANZATO' di cui dobbiamo ancora scegliere le date)

"GIT è QUASI il social per gli informatici :3" cit Renzo Davoli oggi 2023/10/05

Cosa significa oggi UNIX? Linux è UNIX? (domanda su telegram)

UNIX è ciò che è coerente con POSIX (o meglio certificato POSIX).
In questo senso Linux è uno UNIX

BASH base

Qual è la sintassi tipica dei comandi UNIX?

verbo, complementi, complemento oggetto (separato da spazi).

Cosa è un file system? (prima di file system dobbiamo parlare di FILE)

FILE

cosa è? vi ricordate le PRATICHE (commercialista / notarile)?

in inglese è intuitivo

'open a file' --> aprire una pratica

'close a file' --> chiudere una pratica

il FILE SYSTEM

folder --> "il faldone" (tutte le pratiche del anno X)

il file system è un Grafo Diretto Aciclico

(in inglese Direct Acyclic Graph --> DAG

https://en.wikipedia.org/wiki/Directed_acyclic_graph)

ARCHIVIO

UNIX usa il file system come strumento di naming. Cosa significa?

"UNIX usa il file system come strumento di naming."

significa che è un metodo di nomenclatura --> un metodo per dare nomi alle cose

"EVERYTHING IS A FILE"

compreso disco, dispositivi di input/output (tastiera --> keyboard, mouse ecc.)

il fatto che "EVERYTHING IS A FILE" è molto utile perchè così

le system call non hanno niente di specifico per i ...

quindi di 'specifico' (per esempio il disco) ha solo il nome (/dev/vda1).

Cosa sono la home directory e la current working directory? (su telegram)

current dir: informazione specifica di ogni processo, è il nodo del file system che si usa per trovare un file o una directory se il path è relativo

home dir: è la current dir al momento del login. Ci si può ritornare mediante il comando 'cd' senza parametri. Dai pathname si può indicare la home dir con una ~ (tilde): e.g. ~/.bashrc qualunque sia la vostra current dir indica il file .bashrc nella vostra home dir.

Qual è la struttura tipica del file system di UNIX? (su telegram)

Cosa sono i Pathname assoluti e relativi? (su telegram)

pathname assoluti, scandiscono l'albero del file system dalla radice e.g. /etc/passwd (iniziano con /). I pathname relativi non iniziano con / e calcolano la posizione nel file system a partire dalla current dir.

Cosa fanno i comandi `ls`, `echo`, `date`, `cat`, `more`, `less`?

`ls`

```
libera@LIBERA:~$ ls
Desktop  Downloads  Pictures  Templates
Documents Music      Public    Videos
```

`ls -l`

```
libera@LIBERA:~$ ls -l
total 32
drwxr-xr-x 2 libera libera 4096 Oct  5 11:18 Desktop
drwxr-xr-x 3 libera libera 4096 Oct  5 17:10 Documents
drwxr-xr-x 3 libera libera 4096 Oct  5 11:06 Downloads
drwxr-xr-x 2 libera libera 4096 Oct  2 12:50 Music
drwxr-xr-x 3 libera libera 4096 Oct  5 18:53 Pictures
drwxr-xr-x 2 libera libera 4096 Oct  2 12:50 Public
drwxr-xr-x 2 libera libera 4096 Oct  2 12:50 Templates
drwxr-xr-x 2 libera libera 4096 Oct  2 12:50 Videos
```

`ls /dev/vda1`

```
libera@LIBERA:~$ ls /dev/vda1
ls: cannot access '/dev/vda1': No such file or directory
libera@LIBERA:~$ ls /dev
acpi_thermal_rel  dri          kmsg          mapper        nvme0n1p3    rkill         tpmrm0        tty18         tty29         tty4          tty50         tty61         urandom        vcsa1         vcsu5         zero
autofs            drm_dp_aux0  kvm           media0        nvme0n1p4    rtc           tty           tty19         tty3          tty40         tty51         tty62         userfaultfd   vcsa2         vcsu6
block            fb0          log           mei0          nvme0n1p5    rtc0          tty0          tty2          tty30         tty41         tty52         tty63         v4l           vcsa3         vcsu7
btrfs-control    fd           loop0         mem           nvme0n1p6    sgx_provision tty1          tty20         tty31         tty42         tty53         tty7           vcs          vcsa4         vfio
bus              full         loop1         net           nvme0n1p7    sgx_vepc      tty10         tty21         tty32         tty43         tty54         tty8           vcs1         vcsa5         vga_arbiter
char             fuse         loop2         nram          nvme0n1p7    sim           tty11         tty22         tty33         tty44         tty55         tty9           vcs2         vcsa6         vchi
console          hidraw0      loop3         ng0n1         port          snapshot      tty12         tty23         tty34         tty45         tty56         ttyS0          vcs3         vcsa7         vhost-net
core            hidraw1      loop4         null          ppp           snd           tty13         tty24         tty35         tty46         tty57         ttyS1          vcs4         vcsu         vhost-vsock
cpu             hpet         loop5         nvme0         psaux         stderr        tty14         tty25         tty36         tty47         tty58         ttyS2          vcs5         vcsu1         video0
cpu_dma_latency hugepages    loop6         nvme0n1       ptmx          stdin         tty15         tty26         tty37         tty48         tty59         ttyS3          vcs6         vcsu2         video1
cuse            initctl      loop7         nvme0n1p1     pts           stdout        tty16         tty27         tty38         tty49         tty6          uhid           vcs7         vcsu3         watchdog
disk            input        loop-control nvme0n1p2     random        tpm0          tty17         tty28         tty39         tty5          tty60         uinput        vcsa          vcsu4         watchdog0
```

```
libera@LIBERA:~$ ls -l !$
ls -l /dev
total 0
crw-r----- 1 root root 10, 122 Oct  5 15:08 acpi_thermal_rel
crw-r----- 1 root root 10, 235 Oct  5 15:08 autofs
drwxr-xr-x  2 root root 360 Oct  5 15:08 block
crw-rw----  1 root disk 10, 234 Oct  5 15:08 btrfs-control
drwxr-xr-x  3 root root  60 Oct  5 17:08 bus
drwxr-xr-x  2 root root 3600 Oct  5 15:27 char
crw-w----- 1 root tty  5,  1 Oct  5 15:08 console
lrwxrwxrwx  1 root root 11 Oct  5 15:08 core -> /proc/kcore
drwxr-xr-x 10 root root 200 Oct  5 15:08 cpu
crw-r----- 1 root root 10, 123 Oct  5 15:08 cpu_dma_latency
crw-r----- 1 root root 10, 203 Oct  5 15:08 cuse
drwxr-xr-x  9 root root 180 Oct  5 17:08 disk
drwxr-xr-x  3 root root 100 Oct  5 15:08 dri
crw-r----- 1 root root 241,  0 Oct  5 15:08 drm_dp_aux0
crw-rw----  1 root video 29,  0 Oct  5 15:08 fb0
lrwxrwxrwx  1 root root 13 Oct  5 17:08 fd -> /proc/self/fd
crw-rw-rw-  1 root root  1,  7 Oct  5 15:08 full
crw-rw-rw-  1 root root 10, 229 Oct  5 15:08 fuse
```

se `/dev/vda1` non desse l'errore 'No such file or directory' e facessi `cat /dev/vda1` mi verrebbe l'errore 'Permission Denied', ovvero non lo potrei fare pk si potrei leggere il disco ma potrei anche scrivere facendo danni... questo ci fa introdurre il concetto di utenti / root, ma in tanti anni di didattica ho notato che 'confondete root e utenti' con 'modo kernel e modo user' sono cose diverse!

ROOT

UTENTI

vengono entrambi eseguiti in modo utente
sono un astrazione dati del sistema operativo

ROOT è "l'unico utente invenzione del Kernel", ovvero 'root' ha i permessi tutti gli altri non li hanno.


```
libera@LIBERA:~$ ls /
bin  dev  home  initrd.img.old  lib32  libx32  media  opt  root  sbin  sys  usr  vmlinuz
boot  etc  initrd.img  lib  lib64  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz.old
```

/bin --> eseguibili durante boot + librerie

/root --> home di root

/log --> log di sistema

/var --> archivi temporanei x i programmi

/tmp --> archivi temporanei x gli utenti

/var vs /tmp? ad esempio in /var c'è ...

'mailbox degli utenti' di un programma di mail... spooler di stampa...

mentre /tmp è nostro (gli utenti non possono scrivere dentro /var)

da notare: /var/backups

/var/log --> syslog --> demoni sempre attivi nel sistema

/usr

/usr/games

- fortune (storicamente per aforismi, dopo ne hanno fatto una versione con parolacce (per imprecare senza sprecare intelletto (- cowsay (la mucca))

sl --> sapete come è nata la keyboard (tastiera) QWERTY?

è nata per scrivere sulle telescriventi (macchine da scrivere)

e per evitare martelletti che partono pk vicini incespichino per le frequenze + frequenti in inglese...

il risultato? è demenziale per scrivere velocemente e BENE

questo pk c'è il collo di bottiglia del PONTE ENCEFALICO

tra l'emisfero sinistro e l'emisfero destro del cervello...

ovvero ci sono buone probabilità di invertire l'ordine

delle lettere pk c'è un problema di CONCORRENZA.

ma esiste un alternativa a questa tastiera che non ha questo problema?

si esiste la DVORACK (it.wikipedia.org/wiki/Tastiera_semplicata_Dvorak)

(c'è libertà) ma poi è difficile per le persone 'riprogrammarsi'.

sl

sl -a

sl -l

sl -al

nota che con -a chiedono aiuto :3

video, linux day, Enrico Zini, 'metodi di cazzeggio avanzato utenti linux' per caso è questo? <https://youtu.be/L60IzQN9n-k?si=CHv-G280xAJ2BbqG>

(ma ora torniamo a essere seri...)

