

Selezione pubblica, per titoli ed esami, per il reclutamento di una unità di personale da assumere con contratto di lavoro a tempo determinato e pieno di categoria C, posizione economica C1, area tecnica, tecnico-scientifica ed elaborazione dati, della durata di 36 mesi, presso il Dipartimento di Matematica dell'Università degli Studi di Roma "Tor Vergata", nell'ambito del Progetto: "Sistemi operativi e server – Progetto Dipartimento di eccellenza"

D.D. n. 1166 del 15/05/2019 – Scadenza 14/06/2019 – Rif. 1459

### PROVA SCRITTA – testo n. 2

Il presente tema d'esame si compone di due parti: un questionario sulla legislazione universitaria e un problema sulla gestione di un sistema operativo.

**Richieste, raccomandazioni e regole.** Il/la candidato/a restituisca il questionario a pagina 2, debitamente compilato. Inoltre, a lui/lei si richiede di scrivere sul/i foglio/i ricevuto/i inizialmente tutte le istruzioni, che traducono in linguaggio di programmazione la procedura descritta a partire dalla pagina 3. Il linguaggio (*PHP*, *C*, *bash scripting*, etc.) è a scelta del/la candidato/a, ma verrà esplicitamente valutata la fattibilità della soluzione adottata dal/la candidato/a medesimo/a.

Si noti che, al termine della descrizione dell'algoritmo, sono allegate alcune importanti informazioni: la struttura tipica di qualche specifico *file* di sistema; le pagine del manuale *online* che sono accessibili da una finestra di terminale in ambiente *GNU/Linux*) e che descrivono alcuni comandi di sistema; etc. Tali contenuti sono inclusi in opportuni riquadri e, laddove è necessario, ad essi ci si riferisce nel testo seguente.

È possibile (anche se non strettamente necessario) aggiungere commenti tra un'istruzione e l'altra, al fine di introdurre delle spiegazioni, qualora il/la candidato/a lo ritenga opportuno. La commissione raccomanda di adottare soluzioni semplici da implementare e da gestire; inoltre, verrà valutata anche la chiarezza con la quale esse sono esposte.

Durante la prova, è consentito l'accesso a manuali e a un *personal computer*, purché esso sia escluso dalla rete. È invece vietato l'utilizzo di telefoni cellulari, *tablet* e altri oggetti che consentono la comunicazione con l'esterno dell'aula d'esame.

La durata della prova scritta è di due ore, al termine delle quali il/la candidato/a dovrà necessariamente consegnare il proprio elaborato.

### **Questionario riguardante la legislazione universitaria**

- [1] Al Credito Formativo Universitario corrispondono:
- ☐ 25 ore di impegno complessivo per studente;
  - ☐ 60 ore di impegno complessivo per studente;
  - ☐ 50 ore di impegno complessivo per studente.
- [2] Quale delle seguenti alternative è corretta?
- ☐ il Rettore presiede il Senato Accademico;
  - ☐ il Rettore presiede il Collegio di disciplina;
  - ☐ il Rettore presiede il Consiglio di Dipartimento.
- [3] Gli obblighi di frequenza agli insegnamenti dei corsi di laurea e di laurea magistrale sono disciplinati da:
- ☐ Statuto di Ateneo;
  - ☐ Regolamento Didattico del corso di studio;
  - ☐ Regolamento Didattico di Ateneo.
- [4] Secondo la disciplina in tema di lavoro universitario, nel comparto università è prevista la possibilità per i dipendenti di ottenere un'aspettativa?
- ☐ Sì, ma solo se lo consente il Regolamento d'Ateneo concernente l'amministrazione del personale;
  - ☐ No, mai;
  - ☐ Sì, a determinate condizioni.

## Problema

Scrivere un programma che, nell'ambito di un *server* di tipo *GNU/Linux*, gestisce gli *e-mail* nella cosiddetta *INBOX*. Affinché tale programma possa essere eseguito anche da un utente normale, **non** potranno essere utilizzati dei comandi che sono esclusivamente prerogativa dell'amministratore del sistema.

## Descrizione dettagliata dell'algoritmo

Si supponga di avere accesso al *server* con il *login-name nomeutente*. Si ispezionino le *e-mail* in arrivo, che sono contenute nel *file*

*/var/spool/mail/nomeutente*

che deve essere aggiornato, intraprendendo alcune delle azioni descritte ai seguenti punti [1]–[6]; i criteri di scelta di quali operazioni effettuare tra quelle sotto-elencate dipendono dal confronto del suddetto *file* con altri tre, che si chiamano *mittenti.txt*, *blacklist.txt* e *spam\_keywords.txt*; questi ultimi includono, rispettivamente, l'elenco dei mittenti riconosciuti, la lista degli indirizzi *IP* ritenuti inaffidabili e un insieme di parole tipiche che compaiono nei *subject* degli *e-mail* indesiderati.

La struttura tipica di un *file /var/spool/mail/nomeutente* è illustrata dall'esempio riportato nei riquadri delle figure 1–2. L'organizzazione dei dati nei *file mittenti.txt*, *blacklist.txt* e *spam\_keywords.txt* è descritta dagli esempi riportati nei riquadri delle figure 3, 4 e 5 rispettivamente.

- [1] Osservando l'esempio di *file /var/spool/mail/nomeutente*, riportato nei riquadri delle figure 1–2, si può verificare facilmente che la separazione tra un *e-mail* e il successivo, è data dalla sequenza

riga vuota,

riga che comincia con la stringa *From*.

Mentre viene ispezionato il *file /var/spool/mail/nomeutente*, al termine della lettura di ogni singolo *e-mail* ne deve essere stata creata una copia su di un *file* temporaneo, da chiamare, ad esempio,

*currentmail.txt*.

- [2] Il contenuto del *file currentmail.txt* deve essere trasferito su altri *file* così come descritto ai seguenti punti [3]–[5B]. Le decisioni riguardo alle azioni da effettuare dipendono dai seguenti quattro dati, che **devono essere acquisiti** dal suddetto *file*:

il *mittente* *nomemittente* (riportato nella prima riga dell'*e-mail*);

l'indirizzo *IP* (se presente, sta nella riga che comincia con *Received: from*;  
 la *data di ricezione* (nella riga che comincia con *Delivery date:*);  
 il *soggetto* (che ovviamente sta nella riga che comincia con *Subject:*).

- [3] Il primo *e-mail* (che è stato inviato dal sistema), deve essere copiato su di un altro *file* temporaneo, chiamato, ad esempio, *tmpspoolmail.txt*.
- [4] Se *nomemittente* è tra quelli accettati (in altre parole, esso è presente nel *file mittenti.txt*, schematicamente rappresentato nel riquadro di figura 3) e l'indirizzo *IP* **non** è incluso in *blacklist.txt* (vedasi il riquadro di figura 4) allora rimangono solo le ulteriori due sotto-opzioni seguenti.
  - [4A] Se la differenza tra la data attuale e quella di ricezione è superiore a 24 ore, allora tutte le righe del *file currentmail.txt* devono essere copiate in coda al *file* che si chiama *nomemittente* e sta nel *folder /home/nomeutente/mail/*; ovviamente, se non esiste il *file nomemittente* all'interno di quel *folder*, allora esso dovrà essere creato, nell'atto della scrittura di quelle righe.  
 Si tenga presente che ci sono vari metodi per effettuare i suddetti confronti tra le date; uno di questi richiede l'utilizzo dell'istruzione *mktime* insieme al comando *gawk*. Alcune spiegazioni ed qualche esempio di uso combinato di *gawk* e *mktime* sono riportati nei riquadri delle figure 6–7.
  - [4B] Se, invece, la data di ricezione **non precede** di 24 ore quella attuale, allora tutte le righe del *file currentmail.txt* devono essere copiate in coda al *file* che si chiama *tmpspoolmail.txt* e che costituirà il nuovo *INBOX* (si vedano le operazioni finali descritte al punto [6]).
- [5] Se *nomemittente* è tra quelli accettati (in altri termini, esso compare nel *file mittenti.txt*) e l'indirizzo *IP* è **incluso** in *blacklist.txt* allora rimangono solo le ulteriori due sotto-opzioni seguenti.
  - [5A] Se almeno una delle parole riportate nel *file spam\_keywords.txt* (schematicamente rappresentato nel riquadro di figura 5) è presente anche nel *Subject* che fa parte del *file currentmail.txt*, allora tutte le righe che compongono quest'ultimo devono es-

sere copiate in coda al *file* che si chiama **Spam** e sta nel *folder* `/home/nomeutente/mail/`; ovviamente, se non esiste il *file* **Spam** all'interno di quel *folder*, allora esso dovrà essere creato, nell'atto della scrittura di quelle righe.

- [5B] In tutti i casi rimanenti (cioè tali che non ricadono tra le opzioni descritte ai punti [3]–[5A]), allora tutto il contenuto del *file* `currentmail.txt` deve essere copiato in coda al *file* che si chiama **dacontrollare** e sta nel *folder* `/home/nomeutente/mail/`; ovviamente, se non esiste il *file* **dacontrollare** all'interno di quel *folder*, allora esso dovrà essere creato, nell'atto della suddetta operazione di scrittura.

- [6] Il *file* temporaneo `currentmail.txt` deve essere rimosso, al termine dell'esame di tutti gli *e-mail* presenti in `/var/spool/mail/nomeutente`. Successivamente, l'altro *file* temporaneo `tmpspoolmail.txt` deve essere prima sovrascritto a `/var/spool/mail/nomeutente` e poi rimosso.

**Allegati.** Seguono cinque allegati, distribuiti su 7 pagine.

```

From MAILER-DAEMON Fri Jun 7 09:46:24 2019
Date: 07 Jun 2019 09:46:24 +0200
From: Mail System Internal Data <MAILER-DAEMON@jarvis.mat.uniroma2.it>
Subject: DON'T DELETE THIS MESSAGE -- FOLDER INTERNAL DATA
Message-ID: <1559893584@jarvis.mat.uniroma2.it>
X-IMAP: 1153910601 0000081746 $Forwarded $MDNSent
Status: RO

This text is part of the internal format of your mail folder, and is not
a real message. It is created automatically by the mail system software.
If deleted, important folder data will be lost, and it will be re-created
with the data reset to initial values.

From spiderman@jarvis.mat.uniroma2.it Wed May 17 13:25:26 2017
Return-path: <spiderman@jarvis.mat.uniroma2.it>
Envelope-to: paperinik@jarvis.mat.uniroma2.it
Delivery-date: Wed, 17 May 2017 13:25:26 +0200
Received: from spiderman (helo=localhost)
        by jarvis.mat.uniroma2.it with local-esmtp (Exim 4.80)
        (envelope-from <spiderman@jarvis.mat.uniroma2.it>)
        id 1dAx51-00080c-Bd
        for paperinik@jarvis.mat.uniroma2.it; Wed, 17 May 2017 13:25:26 +0200
Date: Wed, 17 May 2017 13:25:23 +0200 (CEST)
From: "Il supereroe di quartiere del Queens" <spiderman@jarvis.mat.uniroma2.it>
To: "Paperino quando e' Paperinik" <paperinik@jarvis.mat.uniroma2.it>
Subject: Re: Altra emergenza x i laboratori
In-Reply-To: <alpine.DEB.2.02.1705171130580.16500@jarvis.mat.uniroma2.it>
Message-ID: <alpine.DEB.2.02.1705171320300.29946@jarvis.mat.uniroma2.it>
References: <alpine.DEB.2.02.1705171130580.16500@jarvis.mat.uniroma2.it>
User-Agent: Alpine 2.02 (DEB 1266 2009-07-14)
MIME-Version: 1.0
Content-Type: TEXT/PLAIN; charset=US-ASCII; format=flowed
jarvis.mat-X-Spam_score: -0.0
jarvis.mat-X-Spam_score_int: 0
jarvis.mat-X-Spam_report: Content analysis details: (-0.0 points, 4.0 required)

pts rule name          description
-----
-0.0 NO_RELAYS          Informational: message was not relayed via SMTP
Status: RO
X-Status:
X-Keywords: $Forwarded
X-UID: 63600

PK, a me risulta che il venerdi' pomeriggio una delle aule 25 e 26
e' occupata da e l'altra aula e' libera.
Mettiti d'accordo con Tony Stark, tony.stark@mit.edu
Spidy

From etabeta@disney.com Wed May 17 14:32:33 2017
Return-path: <etabeta@disney.com>
Envelope-to: paperinik@jarvis.mat.uniroma2.it
Delivery-date: Wed, 17 May 2017 14:32:33 +0200
Received: from smtp.etabeta@disney.com ([x.y.z.23])
        by jarvis.mat.uniroma2.it with esmtps (TLS1.2:DHE_RSA_AES_256_CBC_SHA256
:2$
        (Exim 4.80)
        (envelope-from <etabeta@disney.com>)
        id 1dAy7z-000A72-1e
        for paperinik@jarvis.mat.uniroma2.it; Wed, 17 May 2017 14:32:33 +0200
Received: from adminpc ([x.y.z.158])

```

**Figure 1.** Esempio di *file* `/var/spool/mail/nomeutente` – Ovviamente, quando compaiono sequenze verticali di punti del tipo `:`, si deve intendere che le righe mancanti sono analoghe a quelle precedenti – Pagina 1 di 2

```

by smtp-2015.uniroma2.it (8.14.4/8.14.4/Debian-8) with ESMTTP id v4HCUEu$
for <paperinik@jarvis.mat.uniroma2.it>; Wed, 17 May 2017 14:30:45 +0200
From: "Eta Beta" <etabeta@disney.com>
To: "Paperino quando e' Paperinik" <paperinik@jarvis.mat.uniroma2.it>
References: <005e01d2cef5$0c5b8700$25129500$@it> <alpine.DEB.2.02.1705171225520$
In-Reply-To: <alpine.DEB.2.02.1705171225520.16500@jarvis.mat.uniroma2.it>
Subject: R: Aula multimediale
Date: Wed, 17 May 2017 14:26:05 +0200
Message-ID: <001e01d2cf08$c7577010$56065030$@it>
MIME-Version: 1.0
Content-Type: text/plain;
        charset="iso-8859-1"
Content-Transfer-Encoding: quoted-printable
X-Mailer: Microsoft Office Outlook 12.0
Thread-Index: AdLO+NW44faop1CBS1OSHMOll1lcagAD5q4Q
Content-Language: it
X-Virus-Scanned: clamav-milter 0.99.2 at smtp-2015
X-Virus-Status: Clean
jarvis.mat-X-Spam_score: 0.0
jarvis.mat-X-Spam_score_int: 0

pts rule name                description
-----
-0.0 RP_MATCHES_RCVD        Envelope sender domain matches handover relay doma$
0.0 URIBL_BLOCKED           ADMINISTRATOR NOTICE: The query to URIBL was block$
                             See
                             http://wiki.apache.org/spamassassin/DnsBlocklists#
                             for more information.

Status: RO
X-Status:
X-Keywords:
X-UID: 63602

Caro PK,
purtroppo il test inizia alle 15 e ha una durata di 30 minuti. Alle 15:30
l'aula 17 sara' libera. Mi dispiace crearti problema ma veramente dovresti
aiutarmi, se puoi.
Grazie
Eta

.
.
.

```

**Figure 2.** Esempio di *file* `/var/spool/mail/nomeutente` – Ovviamente, quando compaiono sequenze verticali di punti del tipo `:`, si deve intendere che le righe mancanti sono analoghe a quelle precedenti – Pagina 2 di 2

```
visione@mat.uniroma2.it
tony.stark@mit.edu

.
.
.

topolino@disney.com
qui@disney.com
quo@disney.com
qua@disney.com

.
.
.

pluto@disney.com
etabeta@disney.com

.
.
.

spiderman@queens-college.edu
```

**Figure 3.** Esempio di *file* mittenti.txt – Ovviamente, quando compaiono sequenze verticali di punti del tipo  $\dot{}$ , si deve intendere che le righe mancanti sono analoghe a quelle precedenti.



```
a.b.12.102
c.d.107.44
e.f.78.188
g.h.21.196

.
.
.

u.v.178.125
x.y.12.48
w.z.126.88
```

**Figure 4.** Esempio di *file* `blacklist.txt` – Ovviamente, i simboli a, b, c, ... w, z stanno a rimpiazzare i primi due numeri delle quaterne che costituiscono l'indirizzo *IP* completo e che qui non viene riportato per comprensibili motivi di correttezza - Inoltre, quando compaiono sequenze verticali di punti del tipo  $\dot{}$ , si deve intendere che le righe mancanti sono analoghe a quelle precedenti.

```
diet  
sex  
cialis  
viagra
```

```
.  
.  
.
```

```
save  
risparmia  
fattura
```

```
.  
.  
.
```

**Figure 5.** Esempio di *file* `spam_keywords.txt` – Ovviamente, quando compaiono sequenze verticali di punti del tipo `.`, si deve intendere che le righe mancanti sono analoghe a quelle precedenti.

```

gawk - Date and time calculation functions

gawk has 3 functions to calculate date and time:

    systime
    strftime
    mktime

Let us see in this article how to use these functions:

systime:
    This function is equivalent to the Unix date (date +%s)
    command. It gives the Unix time, total number of seconds
    elapsed since the epoch(01-01-1970 00:00:00).

$ echo | awk '{print systime();}'
1358146640

Note: systime function does not take any arguments.

strftime:
    A very common function used in gawk to format the systime into
    a calendar format. Using this function, from the systime, the
    year, month, date, hours, mins and seconds can be separated.

Syntax:
    strftime (<format specifiers>,unix time);
1. Printing current date time using strftime:

$ echo | awk '{print strftime("%d-%m-%y %H-%M-%S",systime());}'
14-01-13 12-37-45

    strftime takes format specifiers which are same as the format
    specifiers available with the date command. %d for date, %m
    for month number (1 to 12), %y for the 2 digit year number, %H
    for the hour in 24 hour format, %M for minutes and %S for
    seconds. In this way, strftime converts Unix time into a date
    string.

2. Display current date time using strftime without systime:

$ echo | awk '{print strftime("%d-%m-%y %H-%M-%S");}'
14-01-13 12-38-08

    Both the arguments of strftime are optional. When the
    timestamp is not provided, it takes the systime by default.

3. strftime with no arguments:

$ echo | awk '{print strftime();}'
Mon Jan 14 12:30:05 IST 2013

    strftime without the format specifiers provides the output in
    the default output format as the Unix date command.

mktime:
    mktime function converts any given date time string into a
    Unix time, which is of the systime format.

Syntax:
    mktime(date time string) # where date time string is a string
    which contains atleast 6 components in the following order:
    YYYY MM DD HH MM SS

1. Printing timestamp for a specific date time :

$ echo | awk '{print mktime("2012 12 21 0 0 0");}'
1356028200

```

**Figure 6.** Spiegazioni e esempi riguardo all'utilizzo combinato del comando gawk con istruzioni che accedono alla data e all'ora di sistema  
 – Pagina 1 di 2

```

    This gives the Unix time for the date 21-Dec-12.

2. Using strftime with mktime:
$ echo | awk '{print strftime("%d-%m-%Y",mktime("2012 12 21 0 0 0"))}'
21-12-2012

    The output of mktime can be validated by formatting the mktime
    output using the strftime function as above.

3. Negative date in mktime:
$ echo | awk '{print strftime("%d-%m-%Y",mktime("2012 12 -1 0 0 0"))}'
29-11-2012

    mktime can take negative values as well. -1 in the date
    position indicates one day before the date specified which in
    this case leads to 29th Nov 2012.

4. Negative hour value in mktime:
$ echo | awk '{print strftime("%d-%m-%Y %H-%M-%S",mktime("2012 12 3 -2 0 0"))}'
02-12-2012 22-00-00

    -2 in the hours position indicates 2 hours before the
    specified date time which in this case leads to "2-Dec-2012
    22" hours.

```

**Figure 7.** Spiegazioni e esempi riguardo all'utilizzo combinato del comando gawk con istruzioni che accedono alla data e all'ora di sistema  
 – Pagina 2 di 2