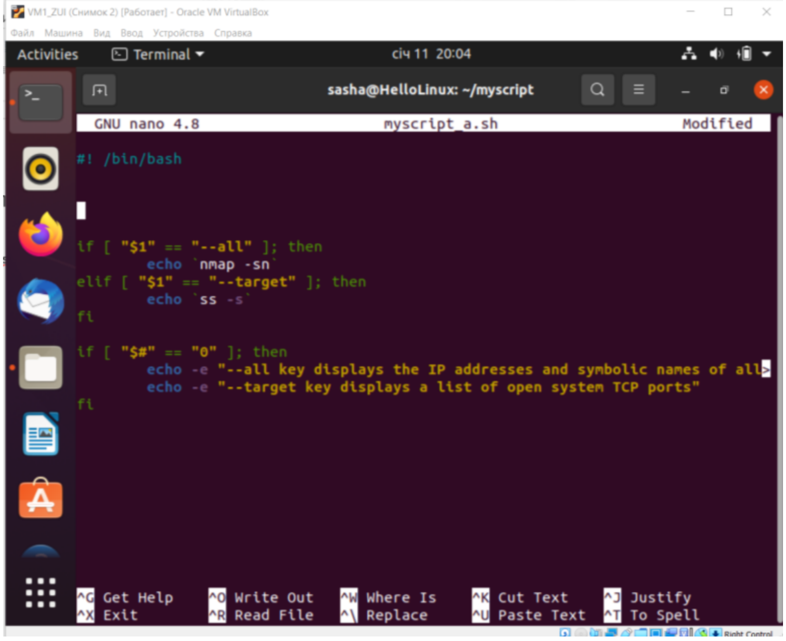
Task 7.1

A

Create a script that uses the following keys: --all; --target.

Nano myscript_a.sh



VM1_ZUI (Снимок 2) [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
Activities
Terminal
січ 11 20:04
sasha@HelloLinux: ~/myscript

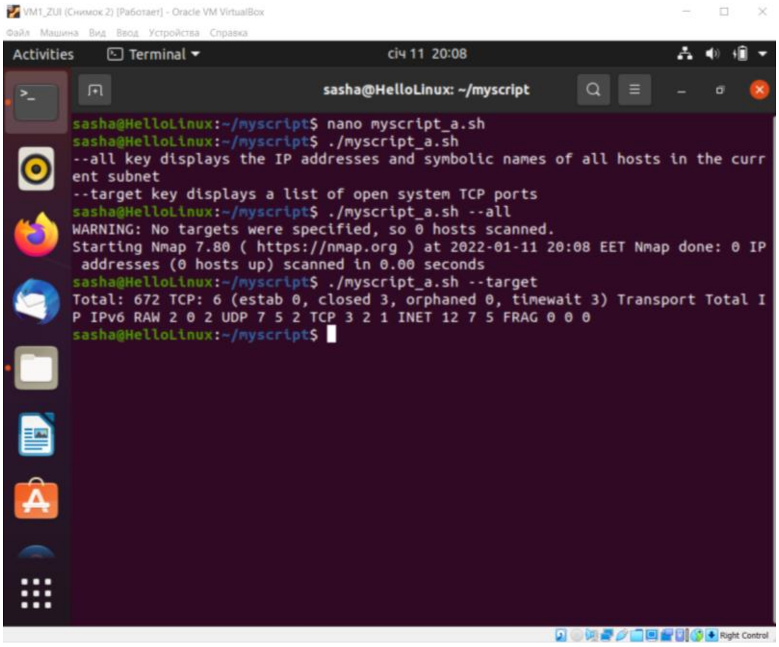
```
  GNU nano 4.8                  myscript_a.sh                    Modified
#! /bin/bash


if [ "$1" == "--all" ]; then
        echo `nmap -sn`
elif [ "$1" == "--target" ]; then
        echo `ss -s`
fi

if [ "$#" == "0" ]; then
        echo -e "--all key displays the IP addresses and symbolic names of all>
        echo -e "--target key displays a list of open system TCP ports"
fi
```

Chmod a+x myscript_a.sh

./myscript_a.sh

./myscript_a.sh –all

./myscript_a.sh – target



VM1_ZUI (Снимок 2) [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
Activities
Terminal
січ 11 20:08
sasha@HelloLinux: ~/myscript

```
sasha@HelloLinux:~/myscript$ nano myscript_a.sh
sasha@HelloLinux:~/myscript$ ./myscript_a.sh
--all key displays the IP addresses and symbolic names of all hosts in the curr
ent subnet
--target key displays a list of open system TCP ports
sasha@HelloLinux:~/myscript$ ./myscript_a.sh --all
WARNING: No targets were specified, so 0 hosts scanned.
Starting Nmap 7.80 ( https://nmap.org ) at 2022-01-11 20:08 EET Nmap done: 0 IP
 addresses (0 hosts up) scanned in 0.00 seconds
sasha@HelloLinux:~/myscript$ ./myscript_a.sh --target
Total: 672 TCP: 6 (estab 0, closed 3, orphaned 0, timewait 3) Transport Total I
P IPv6 RAW 2 0 2 UDP 7 5 2 TCP 3 2 1 INET 12 7 5 FRAG 0 0 0
sasha@HelloLinux:~/myscript$ 
```

B

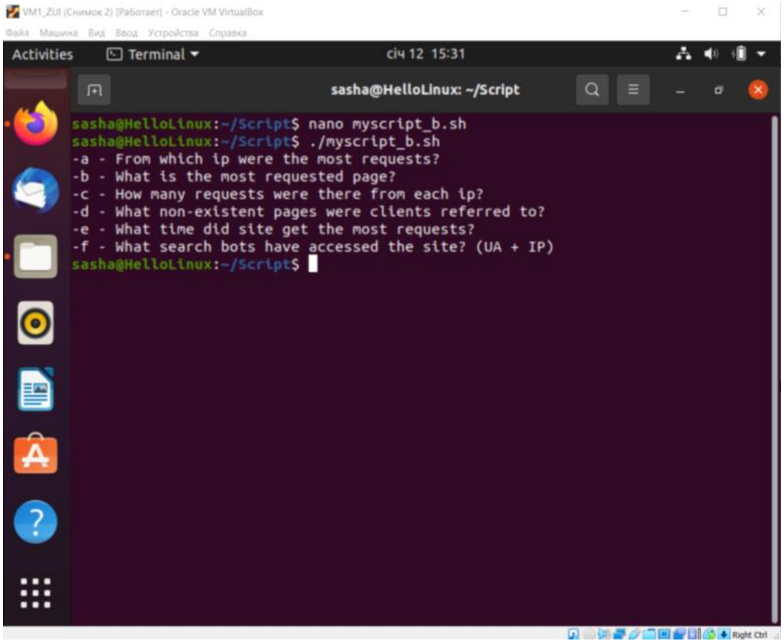
Script text:

```
#! /bin/bash


a=$(less /var/log/apache_logs.txt | grep -E -o '([0-9]{1,3}[\.]){3}[0-9]{1,3}' | sort | uniq -c | sort -gr)
if [ "$1" == "-a" ]; then
        echo $a
fi


b=$(less /var/log/apache_logs.txt | awk '{print $7}' | sort | uniq -c | sort -gr | head -n 5)
if [ "$1" == "-b" ]; then
        echo $b
fi


c=$(less /var/log/apache_logs.txt | grep -E -o '([0-9]{1,3}[\.]){3}[0-9]{1,3}' |awk '{print  " Number of request " $1 " From IP " $2}' | sort | uniq -c | sort -gr)


if [ "$1" == "-c" ]; then
        echo $c
fi


d=$(less /var/log/apache_logs.txt | grep "404" | awk '{print $7}' | head -n 5)


if [ "$1" == "-d" ]; then
        echo $d
fi


e=$(less /var/log/apache_logs.txt |awk '{print $4}' | sort | uniq -c | sort -gr)


if [ "$1" == "-e" ]; then
        echo $e
fi
```

```
f=$(less /var/log/apache_logs.txt | grep "bot" | awk  '{print $7}' | head -n 5)


if [ "$1" == "-f" ]; then
            echo $f
fi


if [ "$#" == "0" ]; then


echo "-a - From which ip were the most requests?"
echo "-b - What is the most requested page?"
echo "-c - How many requests were there from each ip?"
echo "-d - What non-existent pages were clients referred to?"
echo "-e - What time did site get the most requests?"
echo "-f - What search bots have accessed the site? (UA + IP)"
fi
```

1. Run script

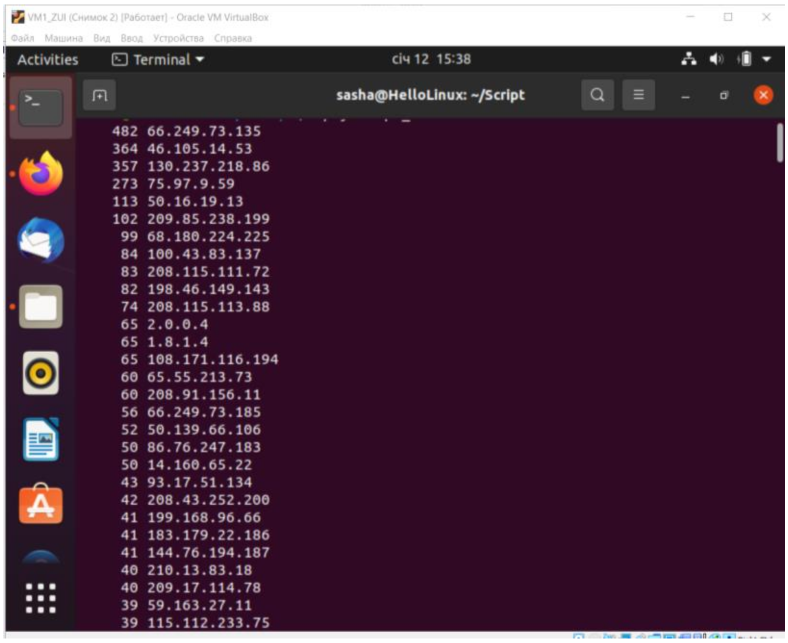
./myscript_b.sh


VM1_ZUI (Снимок 2) [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
Activities
Terminal
січ 12 15:31
sasha@HelloLinux: ~/Script
Right Ctrl

```
sasha@HelloLinux:~/Script$ nano myscript_b.sh
sasha@HelloLinux:~/Script$ ./myscript_b.sh
-a - From which ip were the most requests?
-b - What is the most requested page?
-c - How many requests were there from each ip?
-d - What non-existent pages were clients referred to?
-e - What time did site get the most requests?
-f - What search bots have accessed the site? (UA + IP)
sasha@HelloLinux:~/Script$ 
```

2. From which ip were the most requests?

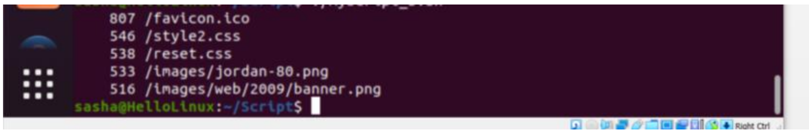
./myscript_b.sh -a


VM1_ZUI (Снимок 2) [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
Activities
Terminal
сiч 12 15:38
sasha@HelloLinux: ~/Script

```
 482 66.249.73.135
 364 46.105.14.53
 357 130.237.218.86
 273 75.97.9.59
 113 50.16.19.13
 102 209.85.238.199
  99 68.180.224.225
  84 100.43.83.137
  83 208.115.111.72
  82 198.46.149.143
  74 208.115.113.88
  65 2.0.0.4
  65 1.8.1.4
  65 108.171.116.194
  60 65.55.213.73
  60 208.91.156.11
  56 66.249.73.185
  52 50.139.66.106
  50 86.76.247.183
  50 14.160.65.22
  43 93.17.51.134
  42 208.43.252.200
  41 199.168.96.66
  41 183.179.22.186
  41 144.76.194.187
  40 210.13.83.18
  40 209.17.114.78
  39 59.163.27.11
  39 115.112.233.75
```

3. What is the most requested page?

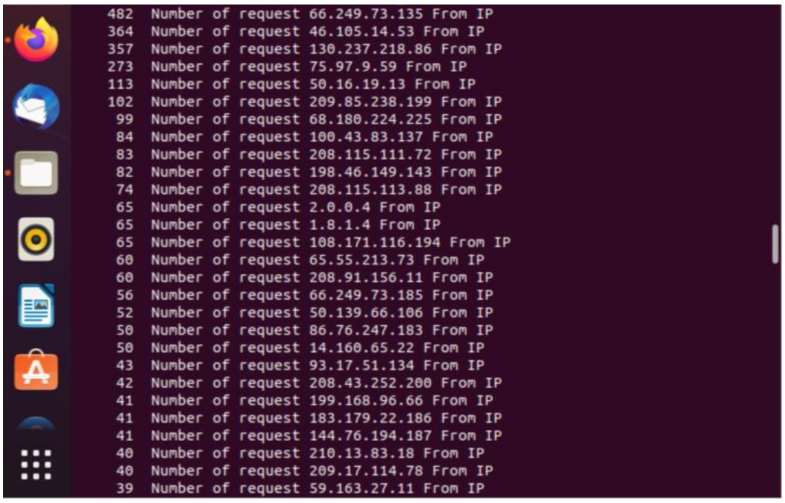
./myscript_b.sh -b



```
 807 /favicon.ico
 546 /style2.css
 538 /reset.css
 533 /images/jordan-80.png
 516 /images/web/2009/banner.png
sasha@HelloLinux:~/Script$
```

4. How many requests were there from each ip?

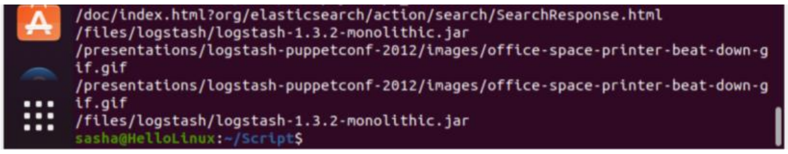
./myscript_b.sh -c



```
 482  Number of request 66.249.73.135 From IP
 364  Number of request 46.105.14.53 From IP
 357  Number of request 130.237.218.86 From IP
 273  Number of request 75.97.9.59 From IP
 113  Number of request 50.16.19.13 From IP
 102  Number of request 209.85.238.199 From IP
  99  Number of request 68.180.224.225 From IP
  84  Number of request 100.43.83.137 From IP
  83  Number of request 208.115.111.72 From IP
  82  Number of request 198.46.149.143 From IP
  74  Number of request 208.115.113.88 From IP
  65  Number of request 2.0.0.4 From IP
  65  Number of request 1.8.1.4 From IP
  65  Number of request 108.171.116.194 From IP
  60  Number of request 65.55.213.73 From IP
  60  Number of request 208.91.156.11 From IP
  56  Number of request 66.249.73.185 From IP
  52  Number of request 50.139.66.106 From IP
  50  Number of request 86.76.247.183 From IP
  50  Number of request 14.160.65.22 From IP
  43  Number of request 93.17.51.134 From IP
  42  Number of request 208.43.252.200 From IP
  41  Number of request 199.168.96.66 From IP
  41  Number of request 183.179.22.186 From IP
  41  Number of request 144.76.194.187 From IP
  40  Number of request 210.13.83.18 From IP
  40  Number of request 209.17.114.78 From IP
  39  Number of request 59.163.27.11 From IP
```

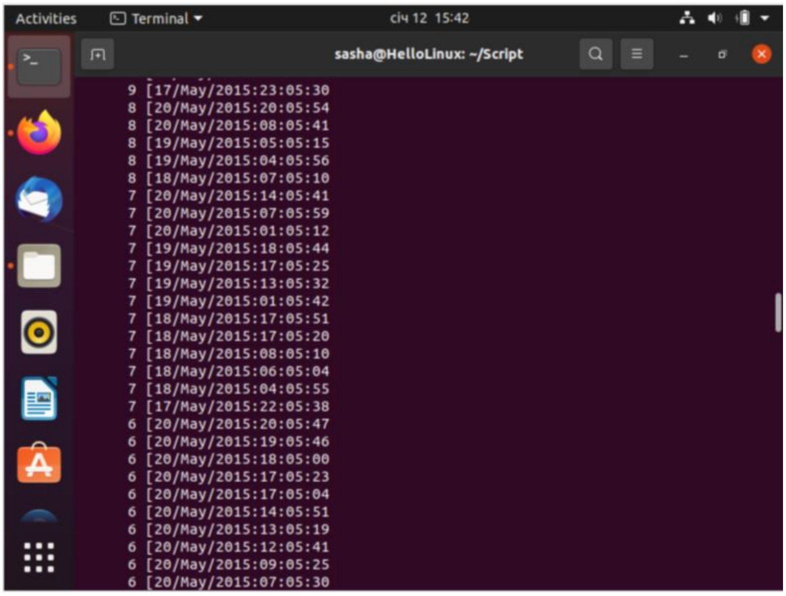
5. What non-existent pages were clients referred to?

./myscript_b.sh -d



```
/doc/index.html?org/elasticsearch/action/search/SearchResponse.html
/files/logstash/logstash-1.3.2-monolithic.jar
/presentations/logstash-puppetconf-2012/images/office-space-printer-beat-down-g
if.gif
/presentations/logstash-puppetconf-2012/images/office-space-printer-beat-down-g
if.gif
/files/logstash/logstash-1.3.2-monolithic.jar
sasha@HelloLinux:~/Script$
```

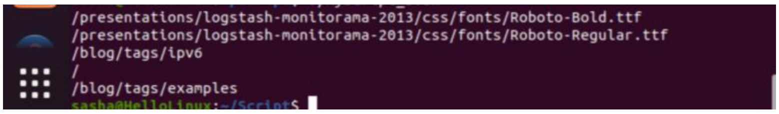
6. What time did site get the most requests?
   ./myscript_b.sh -e



```
  9 [17/May/2015:23:05:30
  8 [20/May/2015:20:05:54
  8 [20/May/2015:08:05:41
  8 [19/May/2015:05:05:15
  8 [19/May/2015:04:05:56
  8 [18/May/2015:07:05:10
  7 [20/May/2015:14:05:41
  7 [20/May/2015:07:05:59
  7 [20/May/2015:01:05:12
  7 [19/May/2015:18:05:44
  7 [19/May/2015:17:05:25
  7 [19/May/2015:13:05:32
  7 [19/May/2015:01:05:42
  7 [18/May/2015:17:05:51
  7 [18/May/2015:17:05:20
  7 [18/May/2015:08:05:10
  7 [18/May/2015:06:05:04
  7 [18/May/2015:04:05:55
  7 [17/May/2015:22:05:38
  6 [20/May/2015:20:05:47
  6 [20/May/2015:19:05:46
  6 [20/May/2015:18:05:00
  6 [20/May/2015:17:05:23
  6 [20/May/2015:17:05:04
  6 [20/May/2015:14:05:51
  6 [20/May/2015:13:05:19
  6 [20/May/2015:12:05:41
  6 [20/May/2015:09:05:25
  6 [20/May/2015:07:05:30
```

7. What search bots have accessed the site? (UA + IP)

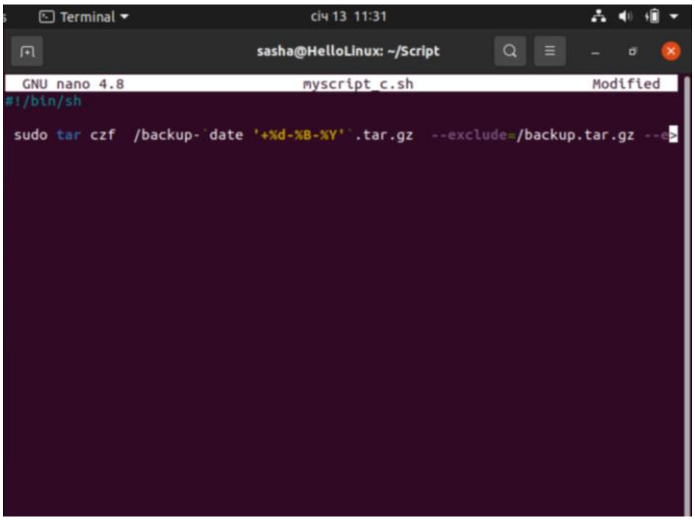
./myscript_b.sh -f



```
/presentations/logstash-monitorama-2013/css/fonts/Roboto-Bold.ttf
/presentations/logstash-monitorama-2013/css/fonts/Roboto-Regular.ttf
/blog/tags/ipv6
/
/blog/tags/examples
sasha@HelloLinux:~/Script$
```

C

Create a script for system backup

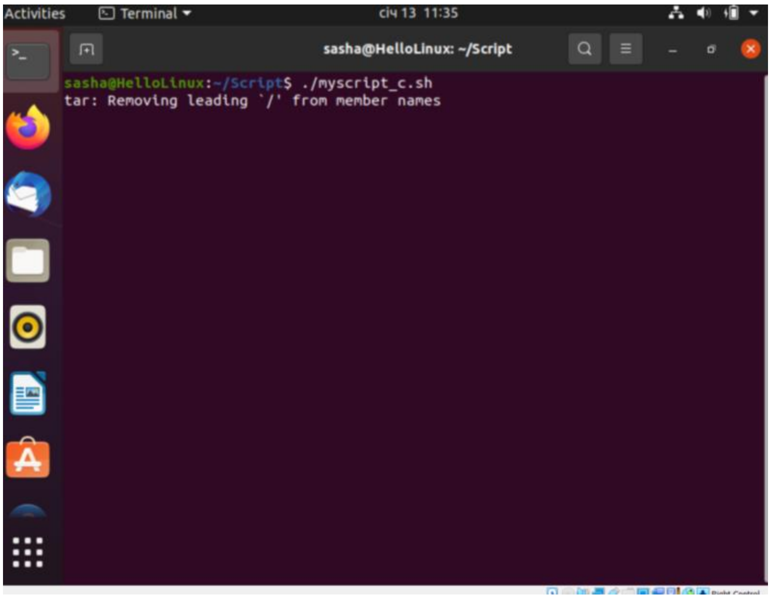


```
  GNU nano 4.8                    myscript_c.sh                    Modified
#!/bin/sh

 sudo tar czf  /backup-`date '+%d-%B-%Y'`.tar.gz  --exclude=/backup.tar.gz --e>
```

Add rights

Chmod a+x myscript_c.sh

./myscript_c.sh

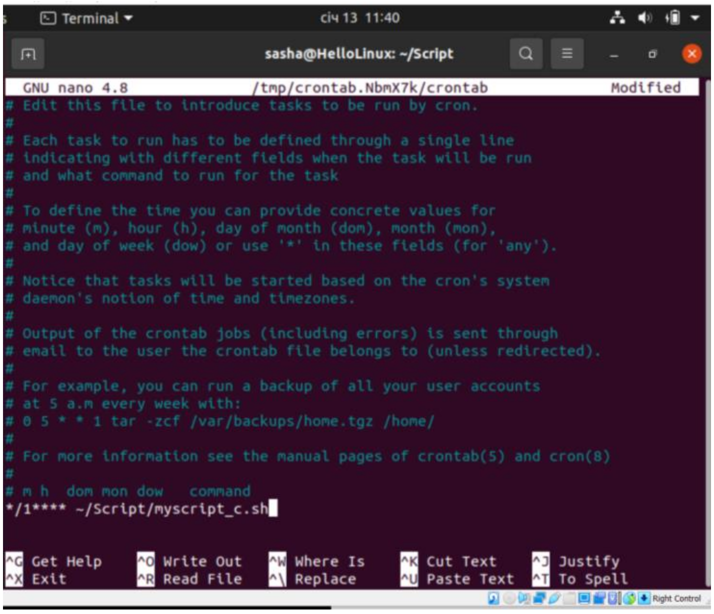


```
sasha@HelloLinux:~/Script$ ./myscript_c.sh
tar: Removing leading `/' from member names
```

A backup is created with the date stored in the root

Add the command to run the script to

crontab with a start frequency of one minute



```
  GNU nano 4.8              /tmp/crontab.NbmX7k/crontab               Modified
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
*/1**** ~/Script/myscript_c.sh


^G Get Help    ^O Write Out   ^W Where Is    ^K Cut Text    ^J Justify
^X Exit        ^R Read File   ^\ Replace     ^U Paste Text  ^T To Spell
```