**1.9.** Which of the following instructions should be allowed only in kernel mode?  
(a) **Disable** all interrupts.  
(b) Read the time-of-day clock. 🡪 user mode  
(c) **Set** the time-of-day dock.  
(d) **Change** the memory heap.

Acd 🡪 kernel mode

Hai mode: kernel và user

Kernel là chế độ hệ điều hành chạy được

User

Phân biệt 2 chế độ mode bit 0 và 1, 0 là kernel mode, 1 là user mode

0 gần vs hệ điều hành, coi là thấp nên gọi là 0, 1 gần vs ng dung hơn, cao hơn nên là 1

**1-12.** Consider a computer system that has cache memory, main memory (RAM) and disk, and the operating system uses virtual memory. It takes 2 nsec to access a word from the cache, 10 nsec to access a word from RAM, and 10 ms to access a word from the disk. If the cache hit rate is 95% and memory hit rate (after a cache miss) is 99%, what is the average time to access a word?

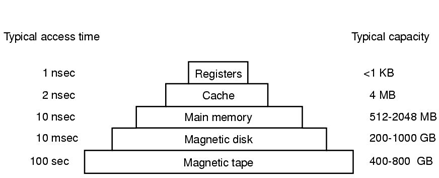
Access a word: làm một công việc gì đó

Tổng thời gian làm xong công việc (access a word) gồm:

* Thời gian dung cache95%
* Thời gian dung Ram+ Disk chiếm 5%
  + Thời gian dung Ram 99%
  + Thời gian dung Disk 1%

Average time =(0.95\*2)+(1-0.95)\*(0.99\*10+(1-0.99))\*10\*106 = 5µs

This is the RAM 0.01 \*disk hit time for disk hr

**1-11 [2nd edition].** An alert reviewer notices a consistent spelling error in the manuscript of an operating systems textbook that is about to go to press. The book has approximately 700 pages, each with 50 lines of 80 characters each. How long will it take to electronically scan the text for the case of the master copy being in each of the levels of memory of Fig. 1-7? For internal storage methods, consider that the access time given is per character, for disk devices assume the time is per block of 1024 characters, and for tape assume the time given is to the start of the data with subsequent access at the same speed as disk access.  


**Figure 1-7.** A typical memory hierarchy. The numbers are very rough approximations.

Tùy từng loại bộ nhớ mà xử lý theo kí tự(per character) hoặc theo khối (per block)

Tổng số ký tự = 700\*50\*80 = 2 800 000

Tổng số khối = 2800000/1024 = 2734.375 = 2735

E.g. download file PDF 1251KB, nếu chỉ down về 1250KB 🡪 file PDF ko mở được

Nếu chỉ dung:

* Register: thời gian mất 2 800 000(ns) hoặc 2.8(ms)
* Cache: Thời gian mất 5.6ms
* RAM mat 28ms
* Magnetic Disk 2735\*10ms = 27350(ms)~ 27s

**Hình trên còn 2 yếu tố:**

* Price: từ trên xuống dưới rẻ dần, càng ở trên càng đắt
* Volatility( có/không mất dữ liệu khi mất điện) phía trên là non-volatility; còn các loại disk là volatility

**2-3.** A computer system has enough room to hold four **(n)**programs in its main memory. These programs are idle waiting for I/O half the time. What fraction of the CPU time is wasted?

Công thức: CPU Utilization=1-p^n, n là số tiến trình, p là % thời gian cho I/O

Theo bài n=4 và p=0,5;=> đs= p^n=6,25

**2-4.** A computer has 2 GB of RAM of which the operating system occupies 256 MB. The processes are all 128 MB (for simplicity) and have the same characteristics. If the goal is 99% CPU utilization, what is the maximum I/O wait that can be tolerated?

0,99=1-p^n=> tính p, tính n=(2x1024-256)/128=14

=>p=72%

**2.33.** Five jobs are waiting to be run. Their expected run times are 9, 6, 3, 5, and *X*. In what order should they be run to minimize average response time? (Your answer will depend on *X*.)

**2.34.** Five batch jobs *A* through *E*, arrive at a computer center at almost the same time. They have estimated running times of 10, 6, 2, 4, and 8 minutes. Their (externally determined) priorities are 3, 5, 2, 1, and 4, respectively, with 5 being the highest priority. For each of the following scheduling algorithms, determine the mean process turnaround time. Ignore process switching overhead.  
(a) Round robin.  
(b) Priority scheduling.  
(c) First-come, first-served (run in order 10, 6, 2, 4, 8).  
(d) Shortest job first.  
For (a), assume that the system is multiprogrammed, and that each job gets its fair share of the CPU. For (b) through (d) assume that only one job at a time runs, until it finishes. All jobs are completely CPU bound.

Với cả 4 thuật toán, việc xác định thứ tự chạy là quan trọng, xác định sai=> tính sai

D, SJF: C=>D=>B=>E=>A

C chạy hết: 2

D: chạy hết 2+4=6

B chạy hết 6+6=12

E chạy hết 12+8=20

A chạy hết 20+10=30

Đs=14

C, thứ tự A=>E 10-6-2-4-8

Đs=10+16+18+22+30 / 5=19,2

b. thứ tự chạy được là 3A 5B 2C 1D 4E

đs= 6+14+24+26+30 / 5=20

**3.4.** Consider a swapping system in which memory consists of the following hole sizes in memory order: 10 KB, 4 KB, 20 KB, 18 KB, 7 KB, 9 KB, 12 KB, and 15 KB. Which hole is taken for successive segment requests of  
(a) 12 KB  
(b) 10 KB  
(c) 9 KB  
for first fit? Now repeat the question for best fit, worst fit, and next fit.

Giải

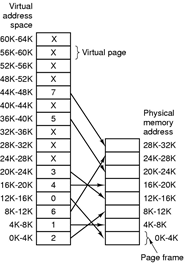
3.1 Yêu cầu (a) --> (b) --> ( c)

- Với thuật toán first-fit: (a) 12KB --> 20KB, (b) 10KB --> 10KB, ( c) 9KB --> 18KB ( Vì 10KB và 20KB đã cấp cho (a) và (b) rồi ).

- Với best-fit: (a) 12KB --> 12KB, (b) 10KB --> 10KB, ( c) 9KB --> 9KB

- Với worst-fit: (a) 12KB --> 20KB, (b) 10KB --> 18KB, ( c) 9KB --> 15KB

- Với next-fit: (a) 12KB --> 20KB, (b) 10KB --> 18KB, ( c) 9KB --> 9KB

**3-6.** Using the page table of Fig. 3-9, give the physical address corresponding to each of the following virtual addresses:  
(a) 20  
(b) 4100  
(c) 8300  


Giải

1. Địa chỉ ảo 20 có địa chỉ vật lí tương đương ở đâu?

Địa chỉ 20 nằm ở ô đầu tiên ( đánh số 2, ô 0K - 4K ), theo hình vẽ thì ô này tương đương ứng với ô nào 8K-12K bên địa chỉ vật lí

Địa chỉ ảo ( 0K - 4K) Địa chỉ vật lí ( 8K-12K)

1. 8192

1 8193

. .

. .

. .

20 8212

21 8213

1. Địa chỉ ảo 4100 có địa chỉ vật lí tương đương ở đâu?

Địa chỉ 4100 nằm ở ô thứ 2 ( đánh số 1, ô 4K - 8K ), theo hình vẽ thì ô này tương đương ứng với ô nào 4K-8K bên địa chỉ vật lí

Địa chỉ ảo ( 4K - 8K) Địa chỉ vật lí ( 4K-8K)

4096 4096

4097 4097

. .

. .

. .

1. 4100
2. Địa chỉ ảo 8300 có địa chỉ vật lí tương đương ở đâu?

Địa chỉ 8300 nằm ở ô thứ 3 ( đánh số 6, ô 8K - 12K ), theo hình vẽ thì ô này tương đương ứng với ô nào 24K-28K bên địa chỉ vật lí

Địa chỉ ảo ( 8K - 12K) Địa chỉ vật lí ( 24K-28K)

8192 24576

8193 24577

. .

. .

. .

1. 24684

**Figure 3-9.** The relation between virtual addresses and physical memory address is given by the page table. Every page begins on a multiple of 4096 and ends 4095 addresses higher, so 4K-8K means 4096-8191 and 8K to 12K means 8192-12287.

**3-11**. A computer with a 32-bit address uses a two-level page table. Virtual addresses are split into a 9-bit top-level page table field, an 11-bit second-level page table field, and an offset. How large are the pages and how many are there in the address space?

**3-16**. A system with 32 bit virtual address. If the page size is 16KB and each table entry occupies 4 bytes, what is the size of the page table?

**3-18**. A machine has 48-bit virtual addresses and 32-bit physical addresses. Pages are 8K. How many entries are needed for a conventional page table? For an inverted page table?

**3-20**. Given the logical address 0xAEF9 (in hexadecimal) with a page size of 256 bytes, what is the page number?

**3-22**. Suppose a virtual address space of 2^32 words and the page size is 2^12 words. If the virtual address is 12345678 in Hexadecimal, what would be the page number in Hexadecimal? (a) 123; (b) 1234; (c) 12345; (d) 123456

**3-24**. Assume that the Page Table below is in effect with the number of lines per page is 400. What is the actual memory location for line 834?

Page Number Page Frame Number

0 8

1 10

2 5

3 11

**3-28.** A computer has four page frames. The time of loading, time of last access, and the R and M bits for each page are as show below (the times are in clock ticks):

Page Loaded Last ref. R M

0 126 280 1 0

1 230 265 0 01

2 140 270 0 0

3 110 285 1 1

(a) Which page will NRU replace?  
(b) Which page will FIFO replace?  
(c) Which page will LRU replace?  
(d) Which page will second chance replace?

Giải

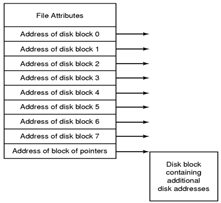
1. Thuật toán NRU ( not recently used), dựa vào các bit R ( referenced) và bit M ( modified), các bit bằng 0 thì càng dễ được chọn --> page 2 sẽ đc chọn làm nạn nhân vì cả hai bit R, M đều =0 ( page này chưa được dùng
2. Thuật toán FIFO (first in first out ) : dựa vào thời gian loaded --> paage 3 (110)
3. Thuật toán LRU ( Least recently used ) ( hệ điều hành hay dùng nhất ) : dựa vào thời gian Last Ref --> page 1 ( 265)
4. Thuật toán second chance: Kết hợp FIFO + NRU

- FIFO --> page 3 nhưng nếu theo NRU thì page 3 có cả bit R, M đều = 1 --> page 3 đc dùng nhiều --> không chọn làm nạn nhân

- FIFO --> page 0 nhưng nếu theo NRU thì page 0 có bit R= 1 --> page 0 đc dùng --> không chọn làm nạn nhân

FIFO --> page 2 nếu theo NRU thì page 3 cả bit R, M đều = 0 --> chọn làm nạn nhân

**4-14.** Consider the i-node shown in Fig. 4-13. If it contains 10 direct addresses of 4 bytes each and all disk blocks are 1024 KB, what is the largest possible file?



**Figure 4-13.** An example inode.

-Địa chỉ trực tiếp(direct): 10

- Địa chỉ gián tiếp=1024KB/4=256

Tổng số địa chỉ=10+256=266, mỗi địa chỉ trỏ đến 1 block 1024KB 🡪 dung lượng file lớn nhất cỏ thể=266\*1024KB =272384 ~ 270MB

**4-29.** A UNIX file system has 1-KB blocks and 4-byte disk addresses. What is the maximum file size if i-nodes contain 10 direct entries, and one single, double, and triple indirect entry each?

Số địa chỉ mà mỗi block có =1KB/4B =256

* Địa chỉ trực tiếp: 10
* Địa chỉ gián tiếp một mức (single): 256
* Địa chỉ gián tiếp hai mức (double): 256^2
* Địa chỉ gián tiếp ba mức (triple): 256**^3**

Tổng số địa chỉ =10 +256 +65536+167772216 = 16 843 018(KB) ~16GB

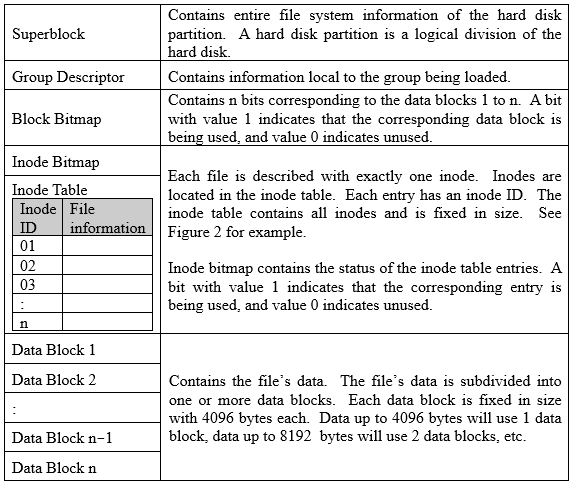
**4-32.** How large is the block size, if the maximum partition size is 128 MB and the FAT type is FAT-16?

Vì định dạng là FAT-16🡪 tổng số địa chỉ= 2^16 =65536 địa chỉ

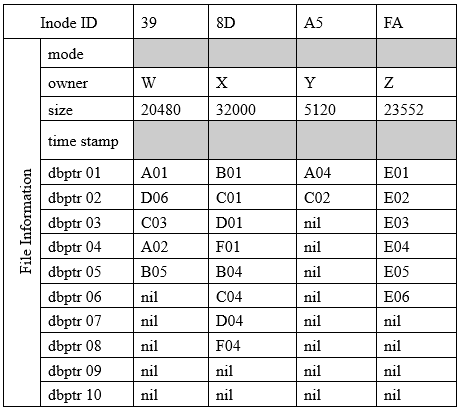
Mà tổng dung lượng phân vùng=128MB🡪 block size =128MB/65536=128\*1024KB/65536= 2 (KB)

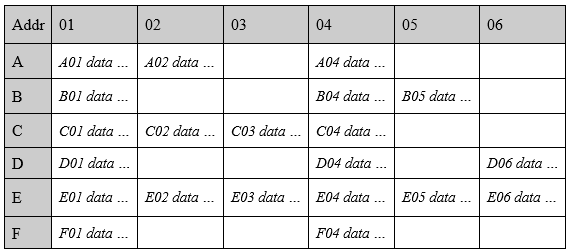
**4-34.** Which FAT type is used, if the maximum partition size is 256 MB and the block size is 4KB?

Tôngr số địa chỉ =256MB/4KB=25\*1024KB/4KB=65536 🡪 FAT-16

**4-40.** A file system manages the files, directories, its storage in the hard disk, and how the files get loaded into memory and used by programs. Each hard disk partition is divided into groups. A group size is ideally in accordance with the structure of the tracks and sectors of the hard disk to make for efficient retrieval. There is no significant performance loss when retrieving data in the same group. Performance loss is noticeable when retrieving data across different groups. Figure 1 shows the structure of a group for some file system. Each group contains a superblock, a group descriptor, a block bitmap, an inode bitmap, an inode table, and finally data blocks.  
  
**Figure 4-41.** Structure of a group.

File information in inode table contains mode (file permission and kind of file), owner information, size, time stamp, and 8 direct pointers (dbptr 01 to 08) to data blocks plus 2 indirect pointers (dbptr 09 and 10) to more data blocks. If a file makes use of 8 data blocks or less, then up to 8 direct pointers will point to the data blocks used. If the file makes use of 9 data blocks or more, each indirect pointer will point to the data block containing the next 1024 direct pointers for succeeding data blocks.

  
**Figure 4-42.** Inode of 4 different files in the same group. Note: “nil” indicates “unused”.

  
**Figure 4-43.** Data blocks of the group in Figure 4-31. The addressing is in row-order form. It begins with A01 through A06 and followed by the next row and so on.

(1) Figure 4-32 represents 4 files found in a group, and Figure 4-33 shows the data blocks of the group. Then, what is the value in hexadecimal that the block bitmap in the group contains?  
a) D26 F25 FE4  
b) D9F 9F9 FE4  
c) E52 ED1 EDC  
d) FA2 F41 EDC

(2) The file system takes into consideration design issue of fragmentation but does not eliminate it. In the example shown in Figure 4-32 and 4-33, internal fragmentation (unused space) will exist where the file with inode (?) will be most fragmented and the file with inode (?) will have ZERO fragmentation.  
a) 39  
b) 8D  
c) A5  
d) FA

(3) Assuming that a series of append file operations are performed in the following order:  
• File of inode 8D doubles in size  
• File of inode A5 increases by 13000 bytes  
• File of inode FA increases by 10000 bytes  
• File of inode 39 increases by 8000 bytes  
The group will run out of data blocks when the file with inode (?) tries to write its updates.  
a) 39  
b) 8D  
c) A5  
d) FA

**5.13.** How much cylinder skew is needed for a 7200-rpm disk with a track-to-track seek time of 1 msec? The disk has 200 sectors of 512 bytes each on each track.

**5.22.** Disk requests come in to the disk driver for cylinders 10, 22, 20, 2, 40, 6, and 38, in that order. A seek takes 6 msec per cylinder moved. How much seek time is needed for  
(a) First-Come, first served.  
(b) Closest cylinder next.  
(c) Elevator algorithm (initially moving upward).  
In all cases, the arm is initially at cylinder 20.

Trong mọi trường hợp, phải xác định đúng thứ tự đi đến các cylinders

1. FCFS: 20🡪 10🡪22🡪20🡪2🡪40🡪6🡪38

số cylinder được đi qua là:

20🡪10: 10

10🡪22: 12

22🡪20: 2

20🡪2: 18

2🡪40: 38

40🡪6: 34

6🡪38: 32

tổng số cylinders=10+12+2+18+38+34+32=146🡺 ĐS: 146\*6=876(ms)

1. Ưu tiên cylinder ở gần nhất trước: 20🡪20🡪22🡪10🡪6🡪2🡪38🡪40

tổng số cylinders=0+ 2+12+4+4+36+2= 60🡺 ĐS: 60\*6=360 (ms)

(c )Thang máy đi lên: 20🡪20🡪22🡪38🡪40🡪10🡪 6🡪2

tổng số cylinders=0+2+16+2+30+4+4=58🡺ĐS: 58\*6=348(ms)

(D) Thang máy có điều chỉnh(Modified Elevator algorithm), đi lên: 20🡪20🡪22🡪38🡪40🡪2🡪6🡪10

Tông số cylinders= 66🡺ĐS: 66\*6=396(ms)

**5-27.** The clock interrupt handler on a certain computer requires 2 msec (including process switching overhead) per clock tick. The clock runs at 60 Hz. What fraction of the CPU is devoted to the clock?

Vì đồng hồ chạy 60HZ 1s(1000ms )sẽ giao động 60 lần. Mà máy tính yêu cầu chạy 2ms/tick

ĐS=2\*60/1000=0.12 hay 12%

**5-28.** A computer uses a programmable clock in square-wave mode. If a 500 Mhz crystal is used, what should be the value of the holding register to achieve a clock resolution of  
(a) a millisecond (a clock tick once every millisecond)?  
(b) 100 microseconds?

Máy tính 500Mh hay 500\*10^6(Hz) =5\*10^8(Hz)🡪 thời gian xử lý =1/5\*10^8(s)=2(ns)

A, A miliseond: 1ms=10^6(ns)=10^3 (ms)🡺 Value=10^6/2=500.000

B, 100 micro=100\*1000=100.000=> Value=100.000/2=50.000

**5-37.** Assuming that it takes 10 nsec to copy a byte, how much time does it take to completely rewrite the screen of an 80 character x 25 line text mode memory-mapped screen? What about a 1024 x 768 pixel graphics screen with 24-bit color?

Mất 10ns để sao chép một byte dữ liệu, nếu sử dụng:

* Text mode screen: tổng số byte dữ liệu=80\*25=2000(bytes)🡪 mất 20.000(ns) hoặc 20. Micro-giây
* Graphic mode screen: tổng số byte dữ liệu=1024\*768\*24/8(bit=>byte)=
* 2 359 296 (bytes)

**6-18.** A computer has six tape drives, with *n* processes competing for them. Each process may need two drives. For which values of *n* is the system deadlock free?

**6.20.** A system has four processes and five allocatable resources. The current allocation and maximum needs are as follows:

Allocated Maximum Available

Process A 1 0 2 1 1 1 1 2 1 3 0 0 x 1 1

Process B 2 0 1 1 0 2 2 2 1 0

Process C 1 1 0 1 0 2 1 3 1 0

Process D 1 1 1 1 0 1 1 2 2 1

What is the smallest value of *x* for which this is a safe state?

QUIZ

1. A CPU may have two or more complete processors, so that can carry out multiple threads in the same time is called:  
   a. Pipeline  
   b. Superscalar  
   c. Multicore  
   d. None of the other choices
2. Examples of general purpose stored program computers include the following except  
   a. Personal computers  
   b. Network servers  
   c. Workstations  
   d. MP3 player
3. Which of the following instructions should be allowed in user mode?  
   a. Disable all interrupts  
   b. Read the time-of-day clock  
   c. Set the time-of-day clock  
   d. Change the memory map
4. As one proceeds down the memory hierarchy (from inboard memory to offline storage), which of the following conditions is correct?  
   a. Decreasing cost per bit  
   b. Decreasing capacity  
   c. Decreasing access time  
   d. None of the other choices
5. Which of the following actions generates an external interrupt?  
   a. An input/output operation is completed.  
   b. A page that does not exist in the main memory is accessed by the virtual storage management.  
   c. A system call instruction is executed.  
   d. Division by zero occurs.
6. What is not a main function of an operating system?  
   a. Provide the users with an extended (virtual) machine  
   b. Manage the I/O devices  
   c. Provide user interfaces  
   d. Support virtual memory
7. What is the main characteristic of real-time operating system?  
   a. Multiple CPU  
   b. Time-sharing  
   c. Time is key parameter  
   d. Many I/O devices
8. What is correct about trap instructions and interrupts?  
   a. Trap instruction switches the execution mode of a CPU from the user mode to the kernel mode  
   b. A trap instruction is caused by a user program to invoke functions in the OS kernel  
   c. An interrupt is caused by an external event  
   d. All of the other choices
9. A simple structuring model for monolithic system includes:  
   a. A main program that invokes the requested service procedure  
   b. A set of service procedures that carry out the system calls  
   c. A set of utility procedures that help the service procedures  
   d. All of the other choices
10. A process where no concurrency inside process; everything happens sequentially is called:  
    a. Random access process  
    b. Sequential process  
    c. Sequential access process  
    d. None of the other choices
11. Which of the following process state transitions is correct, when the external event for which a process was waiting happens?  
    a. Running -> Blocked (waiting)  
    b. Running -> ready  
    c. Blocked (waiting) -> ready  
    d. Ready -> running
12. How many percent of the CPU time is wasted, when a computer system has enough room to hold two program and these programs are idle waiting for I/O half the time?  
    a. 50%  
    b. 25%  
    c. 75%  
    d. None of the other choices

LAB

Một số chú ý khi sử dụng Ubuntu Linux:

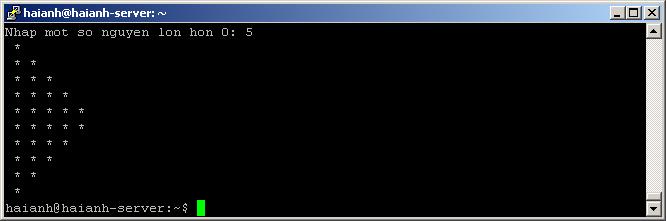
1. Keywords: ubuntu 12.04 repository archive
2. <https://superuser.com/questions/339537/where-can-i-get-the-repositories-for-old-ubuntu-versions>
3. sudo gedit /etc/apt/sources.list
4. precise
5. sudo apt-get update
6. sudo apt-get install netbeans

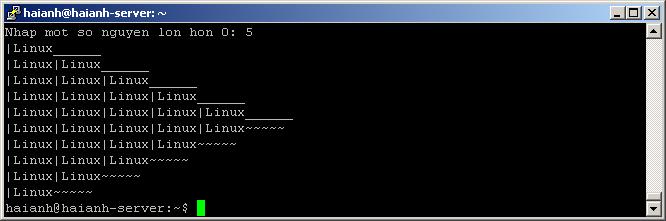
**Basic Linux commands:** <https://github.com/anhnh2/Modern-Operating-Systems/raw/master/os-lab_Linux-commands.pdf>

**Shell programming:** <https://github.com/anhnh2/Modern-Operating-Systems/raw/master/os-lab_Shell_programming.pdf>

Bài 1: Nhập một số từ bàn phím, in ra màn hình tam giác như sau: 

Bài 2: Nhập một số từ bàn phím, in ra màn hình tam giác như sau: 

Bài 3: Nhập một số từ bàn phím, in ra màn hình tam giác đối xứng như sau: 

Bài 4: Nhập một số từ bàn phím, in ra màn hình hình vẽ sau: 

Bài 5: Nhập một số từ bàn phím, in ra màn hình hình vẽ sau: 

Bài 6: Viết một Shell script để tạo ra menu tương tác với người dùng gồm 5 lựa chọn như sau (mỗi lựa chọn được 2 điểm):

[1] Show information about you (name, student ID, class)

[2] Show today date/time

[3] Show all files in current directory

[4] Show calendar

[5] Exit/Stop

Tùy vào lựa chọn của người dùng, chương trình đưa ra thông tin tương ứng, nếu người dùng chọn từ 1-4 thì sau khi thực hiện xong một việc phải quay lại để người dùng chọn chức năng tiếp theo. Chỉ khi ấn số 5 thì chương trình mới thoát.

Bài 7: Sử dụng các lệnh sau để kiểm tra thông tin liên quan đến bộ nhớ (memory):

1. $free -m
2. $cat /proc/meminfo
3. $vmstat -s
4. $top
5. $htop
6. #dmidecode -t 17

*Tham khảo:*[*https://devconnected.com/how-to-check-ram-on-linux/*](https://devconnected.com/how-to-check-ram-on-linux/)*và*[*https://book.hacktricks.xyz/forensics/basic-forensic-methodology/memory-dump-analysis/volatility-examples*](https://book.hacktricks.xyz/forensics/basic-forensic-methodology/memory-dump-analysis/volatility-examples)

Bài 8: Dựa trên bài mẫu “copyfile.c” (*Tham khảo:*[*https://github.com/anhnh2/Modern-Operating-Systems*](https://github.com/anhnh2/Modern-Operating-Systems)) bổ sung các yêu cầu sau cho chương trình:  
(a) Đưa ra thông báo lỗi khi không nhập đủ tên file nguồn, file đích  
(b) Kiểm tra, nếu đối tượng yêu cầu copy không tồn tại hoặc là thư mục thì không copy, và đưa ra thông báo lỗi giống hệt như khi sử dụng lệnh cp trong Linux  
(c) Đặt thiết lập các quyền permission cho file mới tạo ra như sau: rw-rw-rw-

Bài 9: Lần lượt chạy các lệnh sau:

1. winsat disk -drive d
2. diskspd -d120 -W10 -C10 -c512M -t4 -o4 -b8k -L -r -Sh -w50 D:\io.dat