PROJECT - NGUYÊN LÝ HỆ ĐIỀU HÀNH

Học kỳ 1 – 2023-2024

Máy ảo sử dụng: Ubuntu VM

Cách thức nộp bài: 01 folder tên mã số sinh viên (***MSSV***) và họ tên của anh chị ***MSSV_HọTên*** chứa

- 01 báo cáo mô tả chi tiết cách giải quyết các vấn đề nêu lên trong project (**kể cả hình ảnh minh họa kết quả và các lệnh** anh/chị thực hiện theo đúng trình tự)
- Tất cả code (cả mức kernel và user) có liên quan

Lưu ý:

- File báo cáo cần có Họ tên và MSSV và phần tuyên bố như bên dưới
- Code cần có chú thích rõ ràng, đầy đủ

Tuyên bố: Project này là do chính tôi, *Quách Minh Hớn* (MSSV: B2110078), tự thực hiện, không sao chép của bất kỳ ai. Nếu có bất cứ sao chép nào, tôi hoàn toàn chịu trách nhiệm.

PHẦN 1: XÂY DỰNG LINUX KERNEL

Thực hiện ở chế độ người dùng root (root user)
$su -

A. CHUẨN BỊ LINUX KERNEL CODE (phần hướng dẫn dưới đây thực hiện trên CentOS. Tuy nhiên, sinh viên cần thực hiện trên Ubuntu)

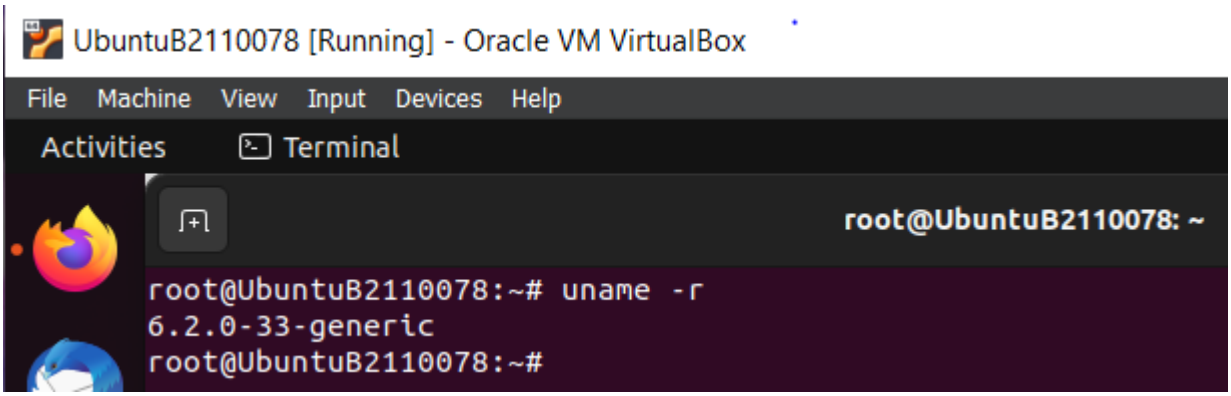
1. Download và cài đặt tool cần thiết vào hệ thống

Nếu thực hiện ở CentOS:
#yum install -y gcc ncurses-devel make wget
#yum apt-get install -y gcc openssl-devel
#yum apt-get install -y gcc elfutils-libelf-devel
#yum apt-get install bison
#yum apt-get install flex

Nếu thực hiện ở Ubuntu:
$sudo apt-get install -y gcc libncurses5-dev make wget
$sudo apt-get install -y gcc libssl-dev
$sudo apt-get install bison
$sudo apt-get install flex

2. Xác định phiên bản hiện tại của kernel:

 # uname –r
6.2.0-33-generic

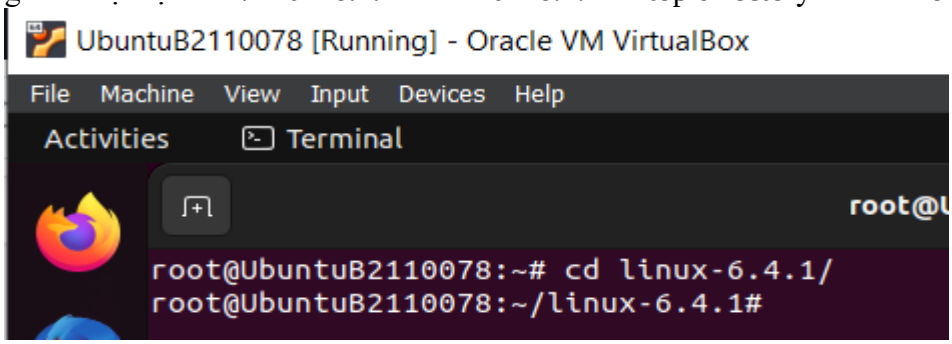


3. Truy cập http://kernel.org hoặc https://cdn.kernel.org/pub/linux/kernel/ và download source code của kernel hiện tại. Kế tiếp, download kernel 5.9 và giải nén source code:

 # wget http://www.kernel.org/pub/linux/kernel/v6.x/linux-6.4.1.tar.xz
 # tar xf linux-6.4.1.tar.xz

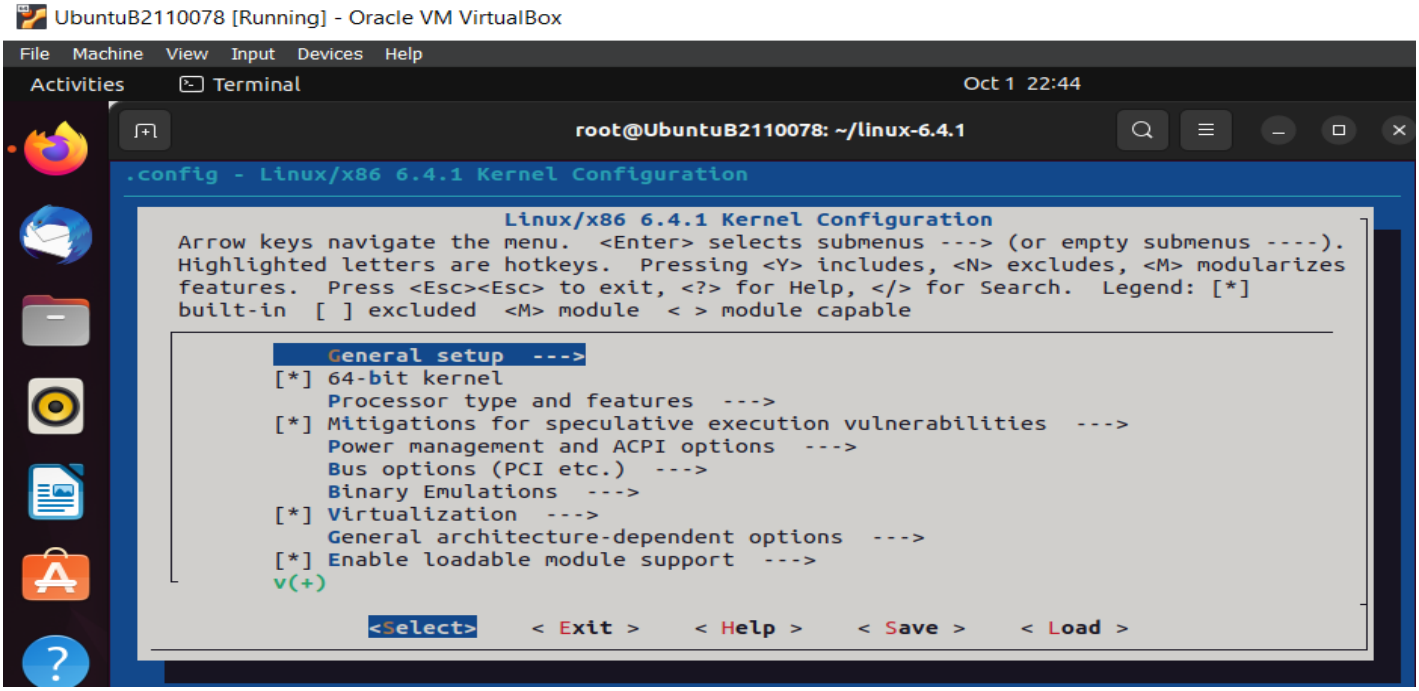
B. CẤU HÌNH KERNEL MỚI

1. Đảm bảo đường dẫn hiện tại ở **~/linux-6.4.1** và "**linux-6.4.1**" ở top directory của kernel source.



2. Tạo file cấu hình (config file)

 # make menuconfig

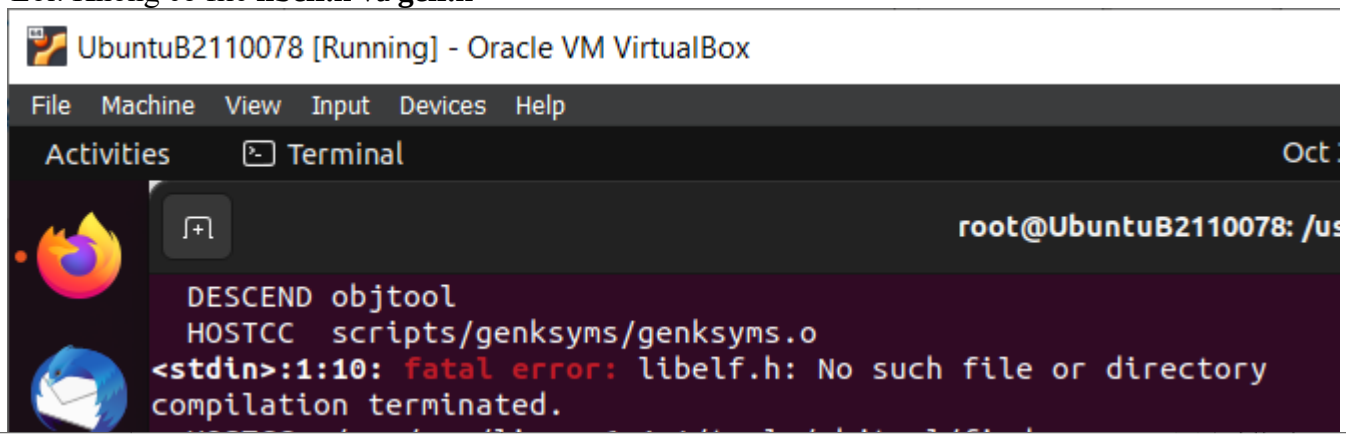


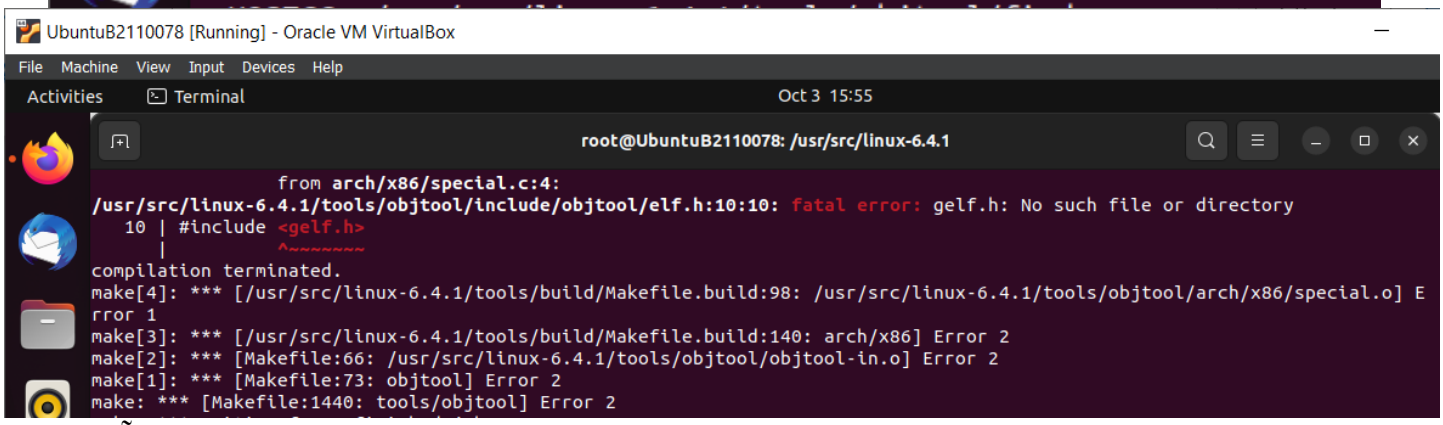
C. BIÊN DỊCH KERNEL

1. Tại **~/linux-5.9**, tạo kernel image nén (compressed kernel image)

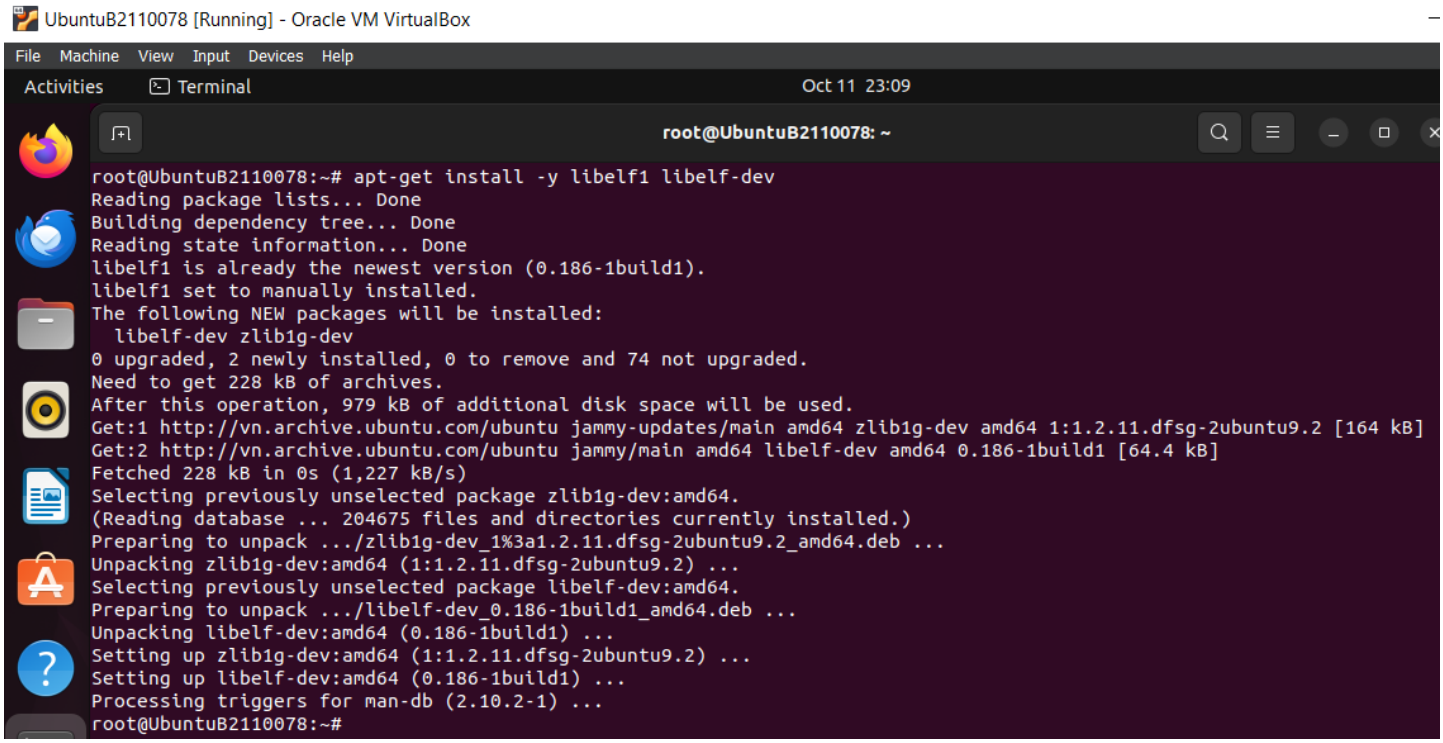
 # make –j4

- Lỗi: Không có file **libelf.h** và **gelf.h**

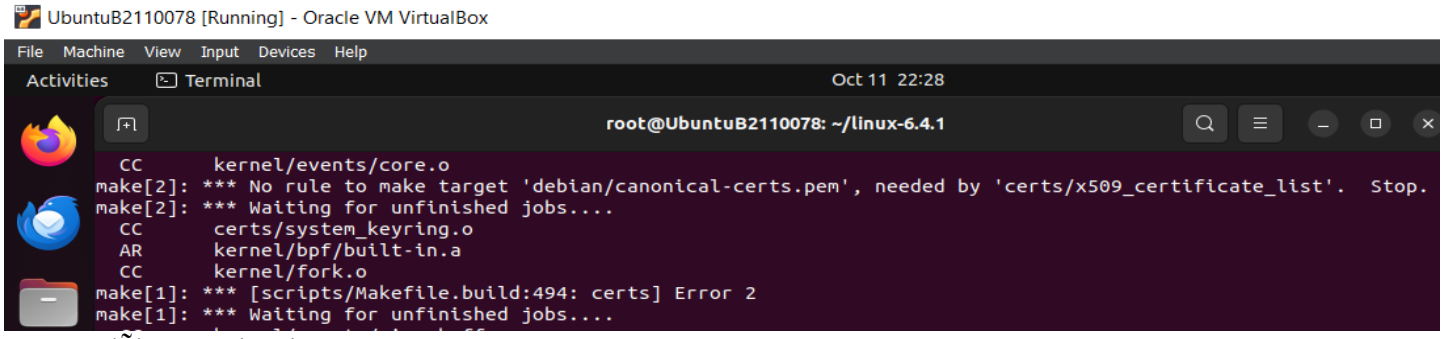




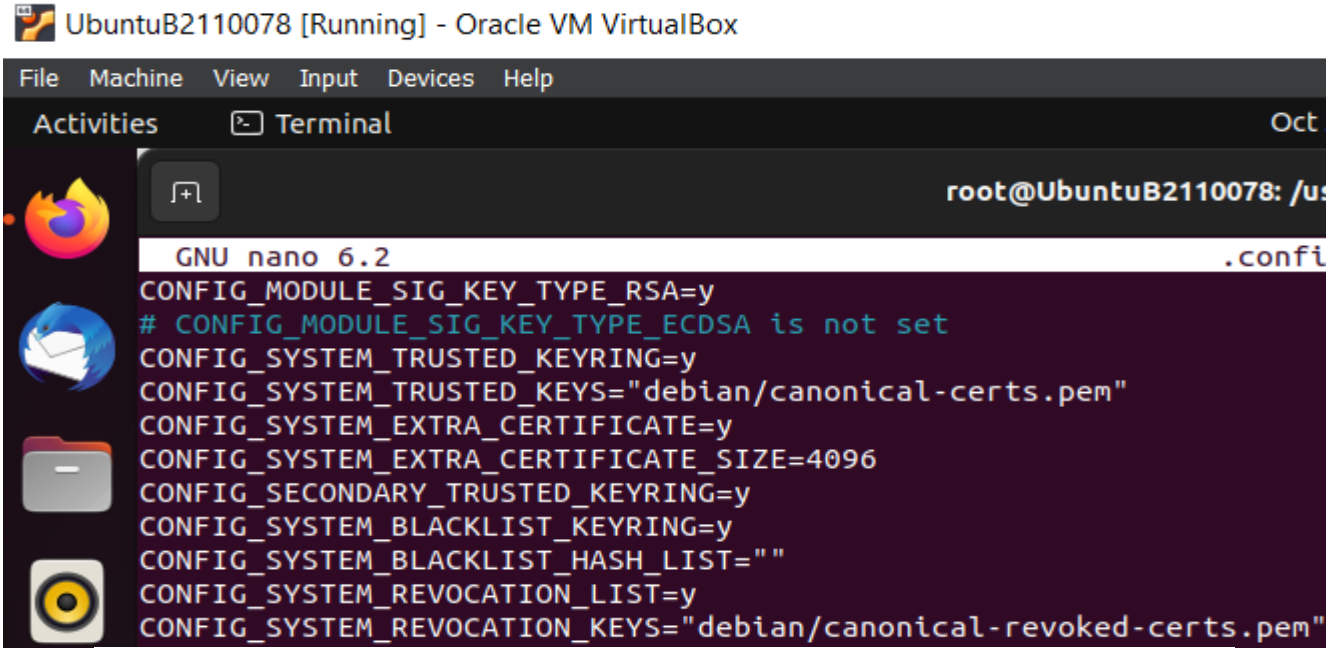
- Sửa lỗi: # **sudo apt-get install -y libelf1 libelf-dev**

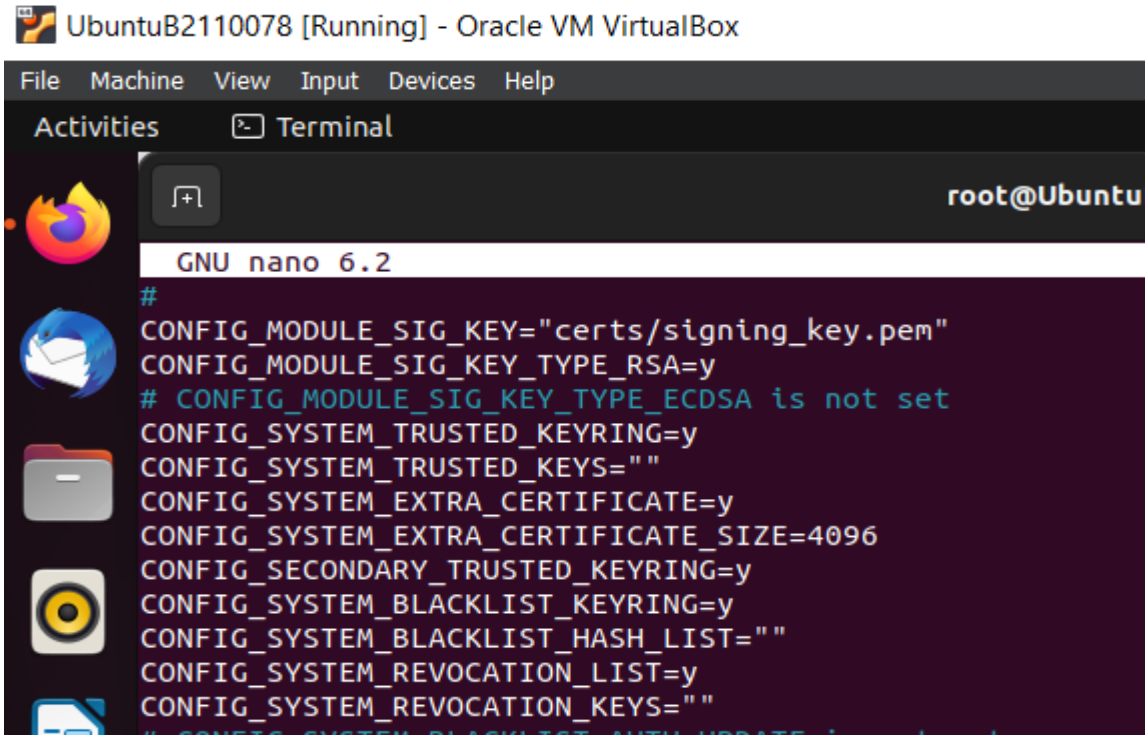


- Lỗi: No rule to make target '**debian/cannonical-certs.pem**'



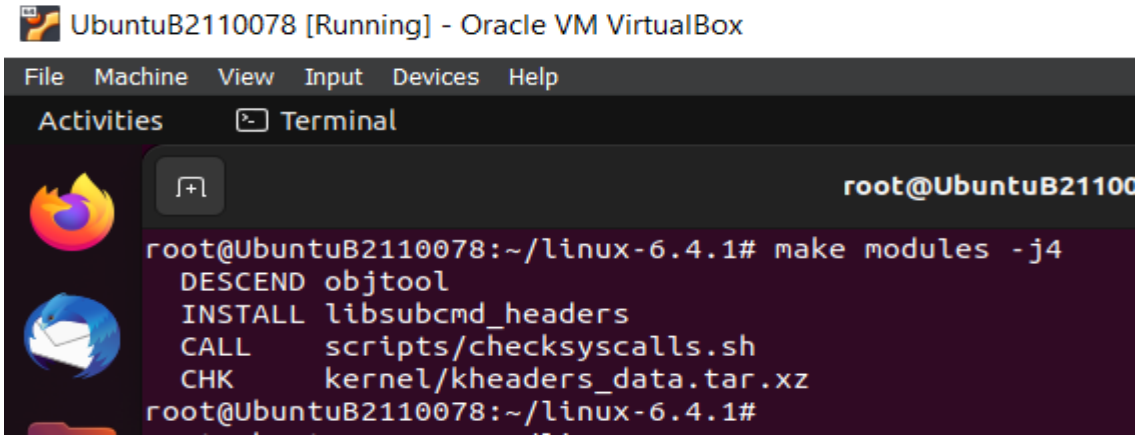
- Sửa lỗi: Xóa giá trị của CONFIG_SYSTEM_TRUSTED_KEYS và CONFIG_SYSTEM_REVOCATION_KEYS





2. Biên dịch kernel modules:

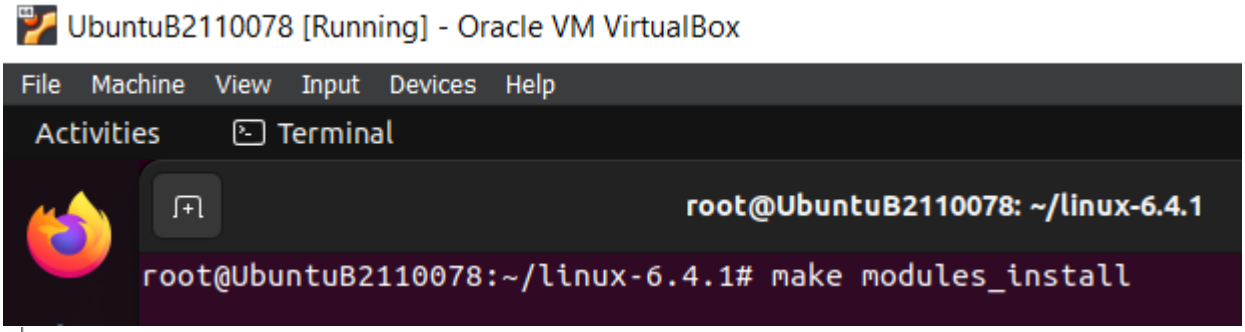
 # make modules

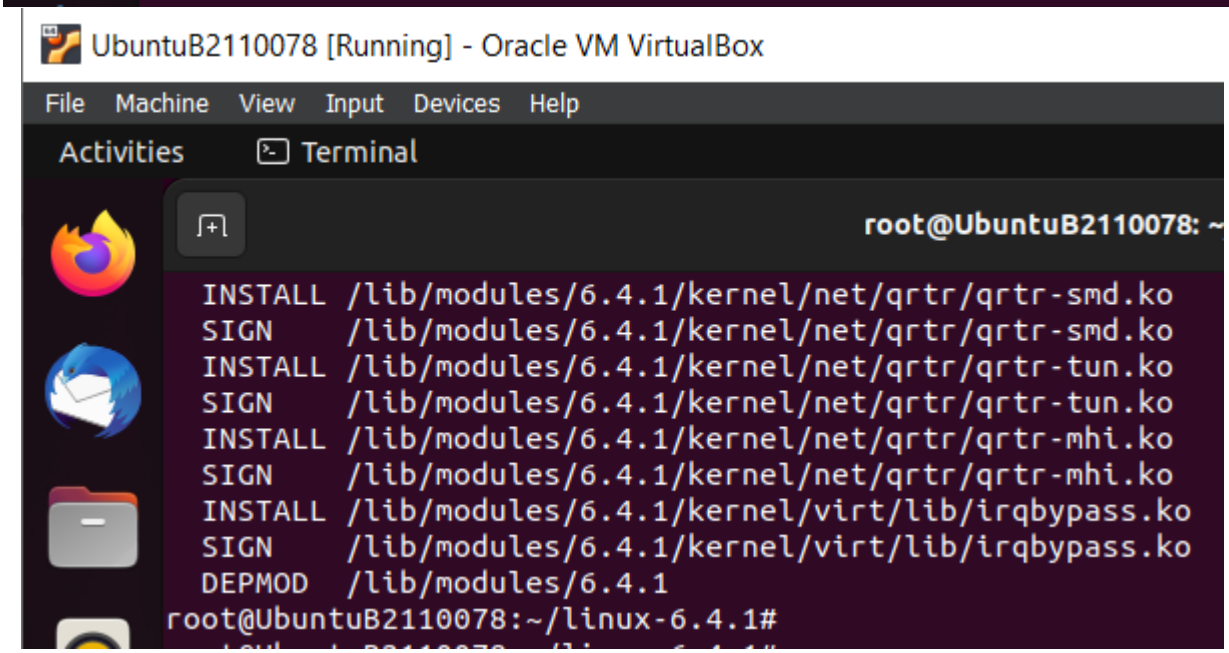


D. CÀI ĐẶT KERNEL

1. Cài đặt kernel modules

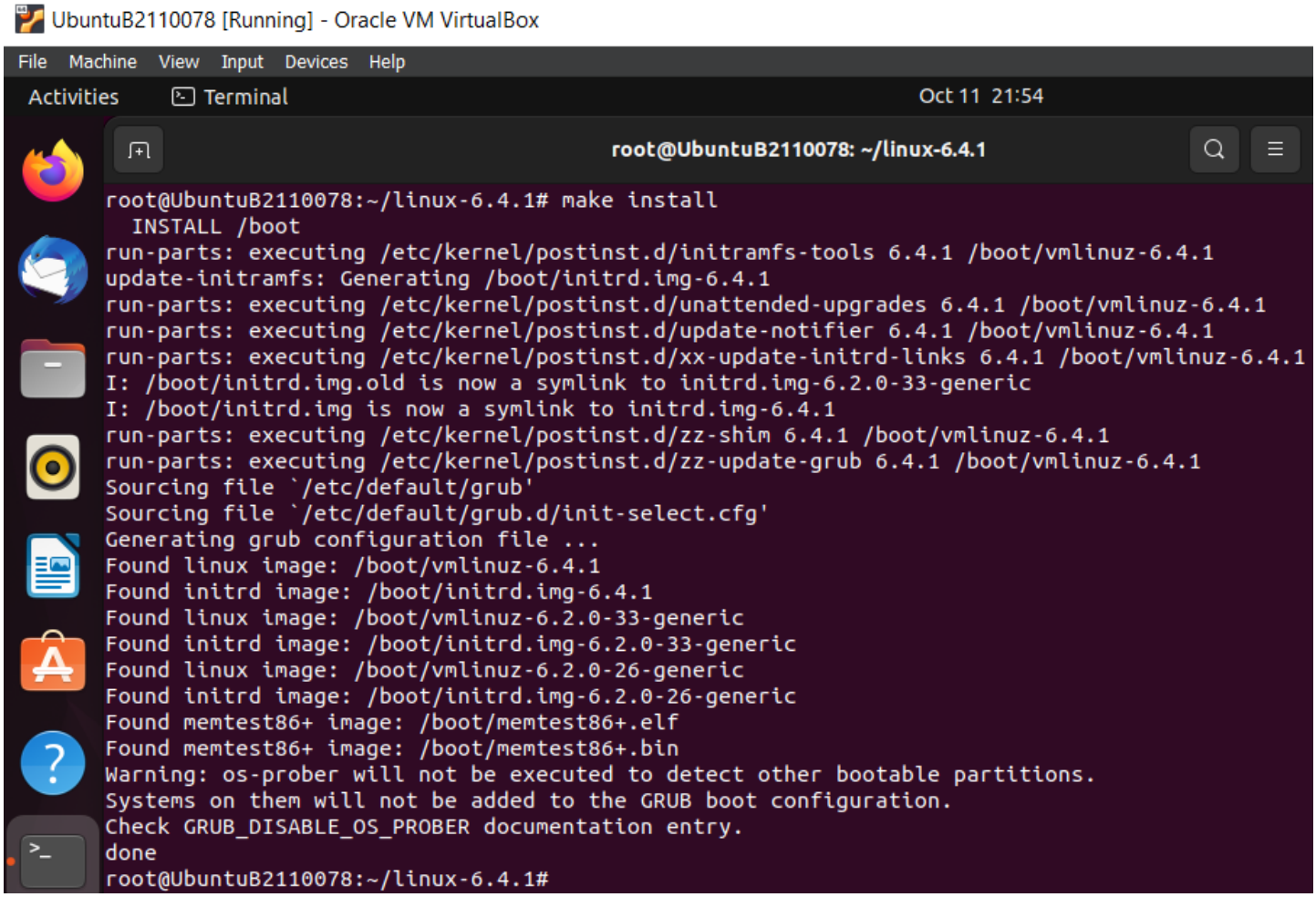
 # make modules_install





2. Cài đặt the kernel

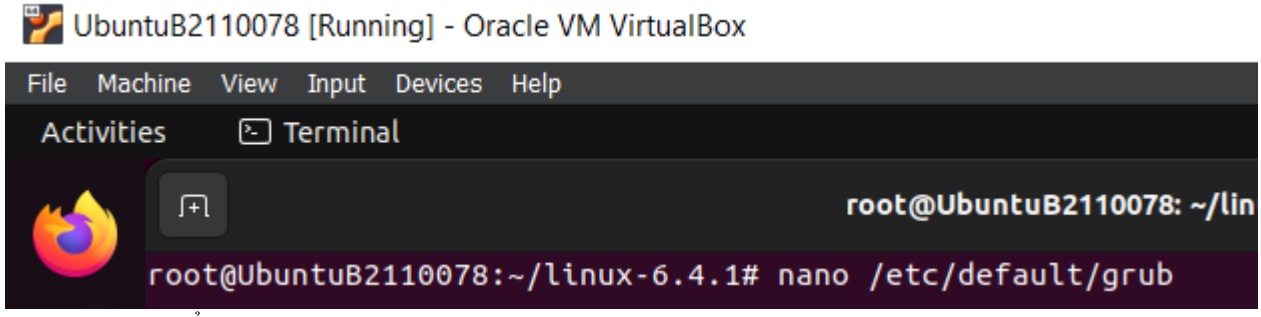
 # make install



E. THAY ĐỔI FILE CẤU HÌNH GRUB (GRUB CONFIGURATION FILE)

Thay đổi file cấu hình grub:

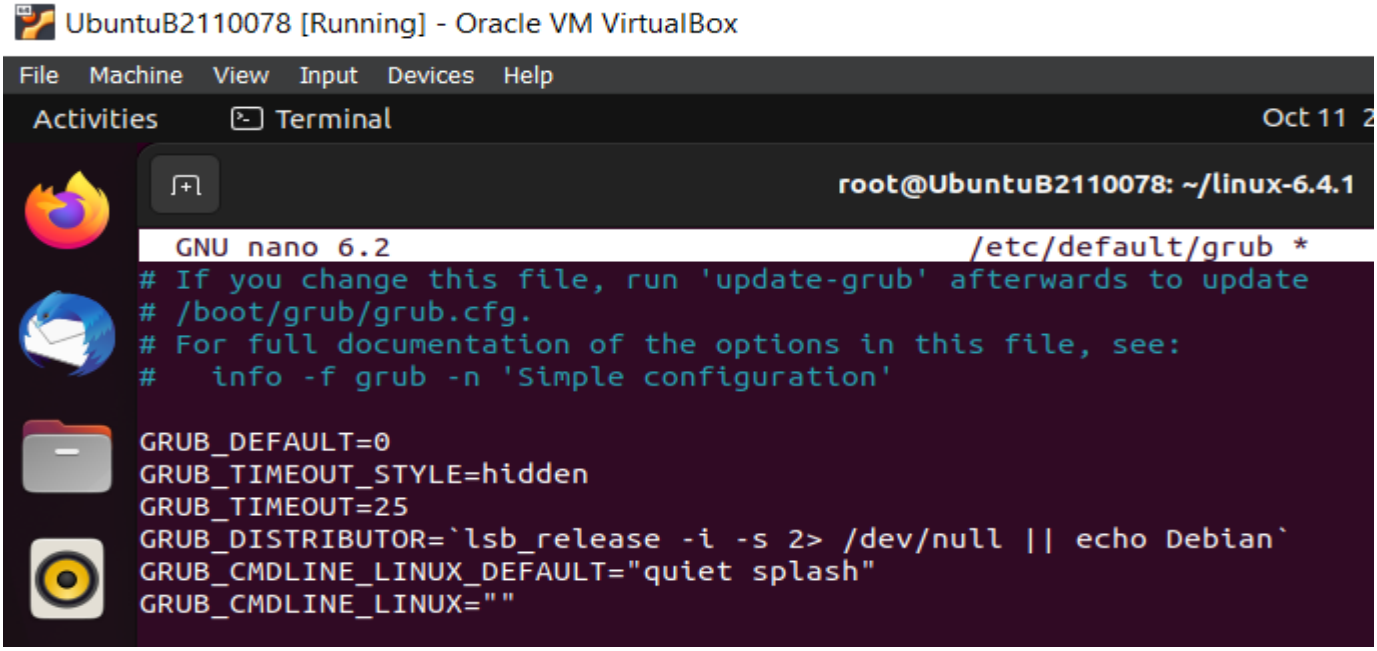
 # vim /etc/default/grub



Thực hiện các thay đổi sau:

GRUB_DEFAULT=0

GRUB_TIMEOUT=25



F. REBOOT VM

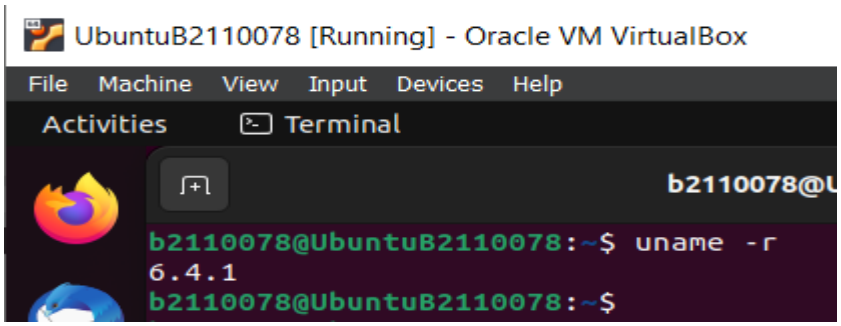
1. Reboot kernel mới:

 # reboot

2. Sau khi boot, kiểm tra thông tin kernel mới có đúng chưa:

 # uname -r

Kết quả: 6.4.1

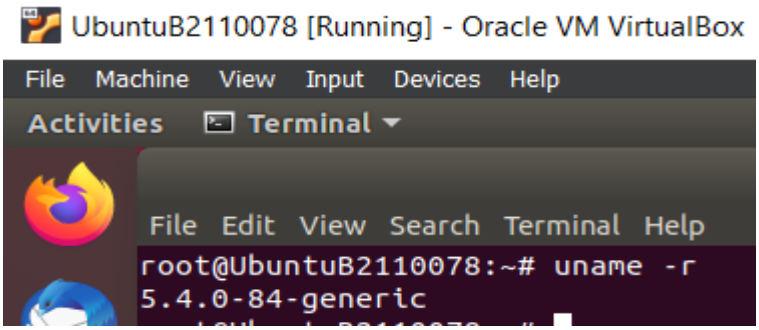


PHẦN 2: THÊM LỜI GỌI HỆ THỐNG VÀO LINUX KERNEL

Thực hiện thêm lời gọi hệ thống **helloworld** vào Linux kernel. Lời gọi hệ thống thực hiện in ra thông điệp "Xin chao. Ten toi la *XXX* " vào syslog *(XXX* là tên và MSSV của anh/chị). Anh/chị cần cài đặt lời gọi hệ thống ở mức kernel và viết một chương trình ở mức người dùng (user-level) để kiểm tra lời gọi hệ thống đã tạo.

A. CHUẨN BỊ KERNEL SOURCE CODE

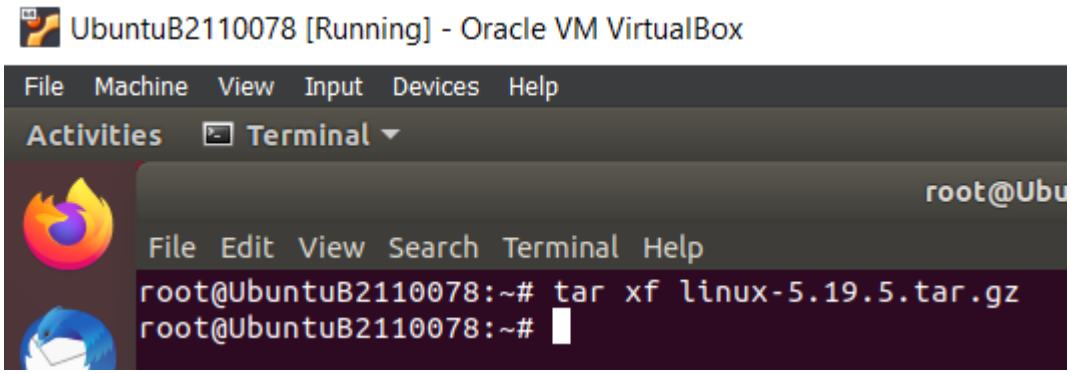
- Xác định phiên bản kernel hiện tại là 5.4.0-84-generic



- Tải xuống kernel phiên bản 5.19.5 và giải nén

 # wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.19.5.tar.gz

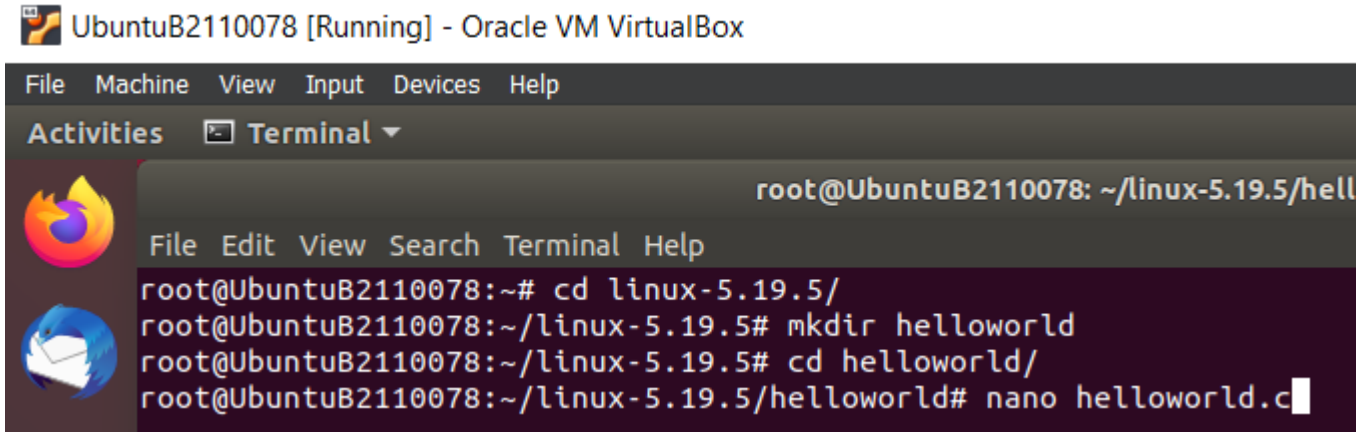
 # tar xf linux-5.19.5.tar.gz

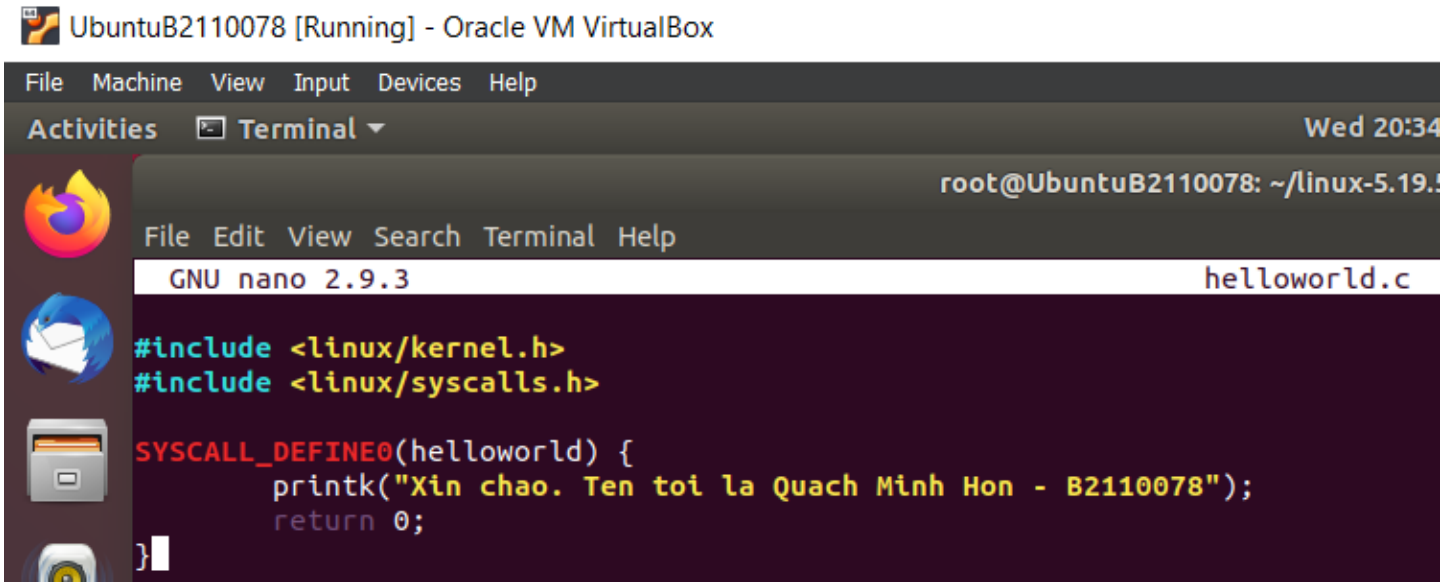


B. THIẾT LẬP SYSCALL HELLOWORLD

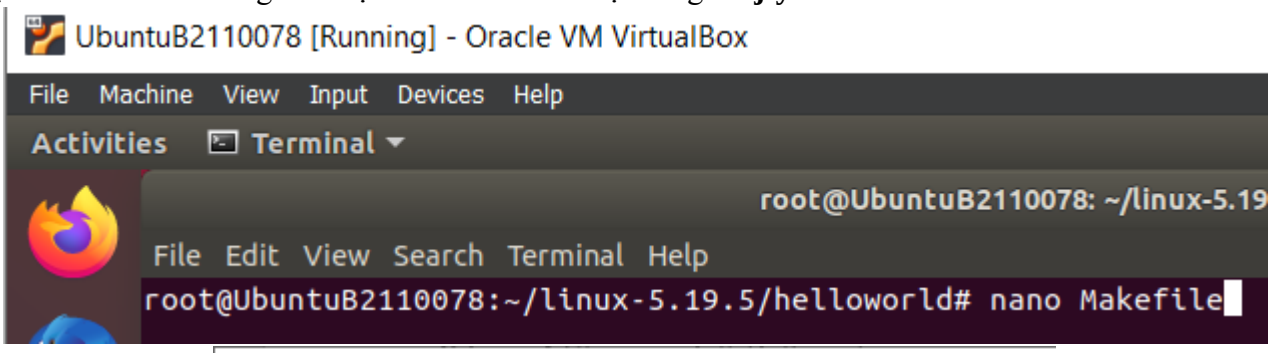
- cd vào thư mục linux-5.19.5, tạo thư mục helloworld
- Trong thư mục helloworld, tạo file helloworld.c với đoạn code bên dưới

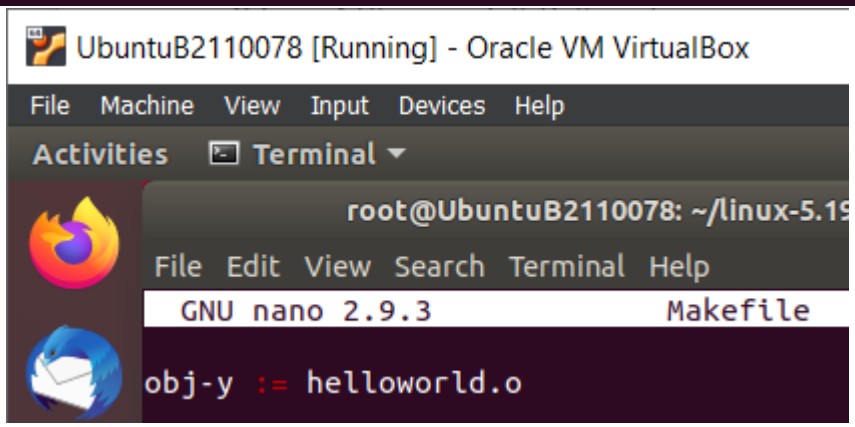
```
#include <linux/kernel.h>
#include <linux/syscalls.h>

SYSCALL_DEFINE0(helloworld) {
        printk("Xin chao. Ten toi la Quach Minh Hon – B2110078");
        return 0;
}
```





- Tạo file Makefile trong thư mục helloworld với nội dung: **obj-y := helloworld.o**



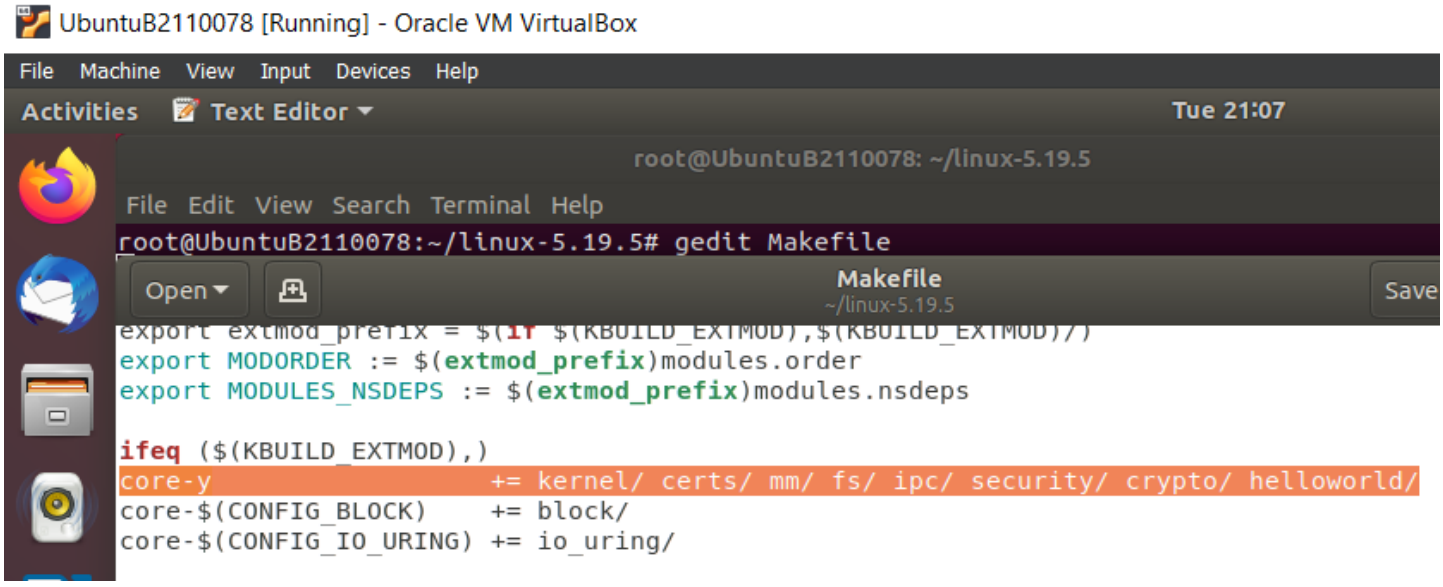


- cd về thư mục linux-5.19.5 để chỉnh sửa file Makefile có sẵn

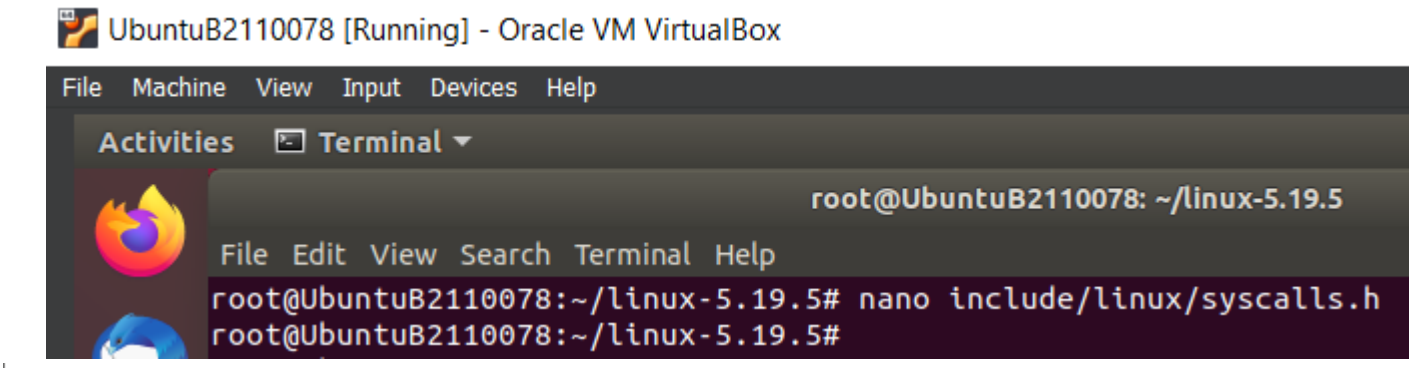
cd ..

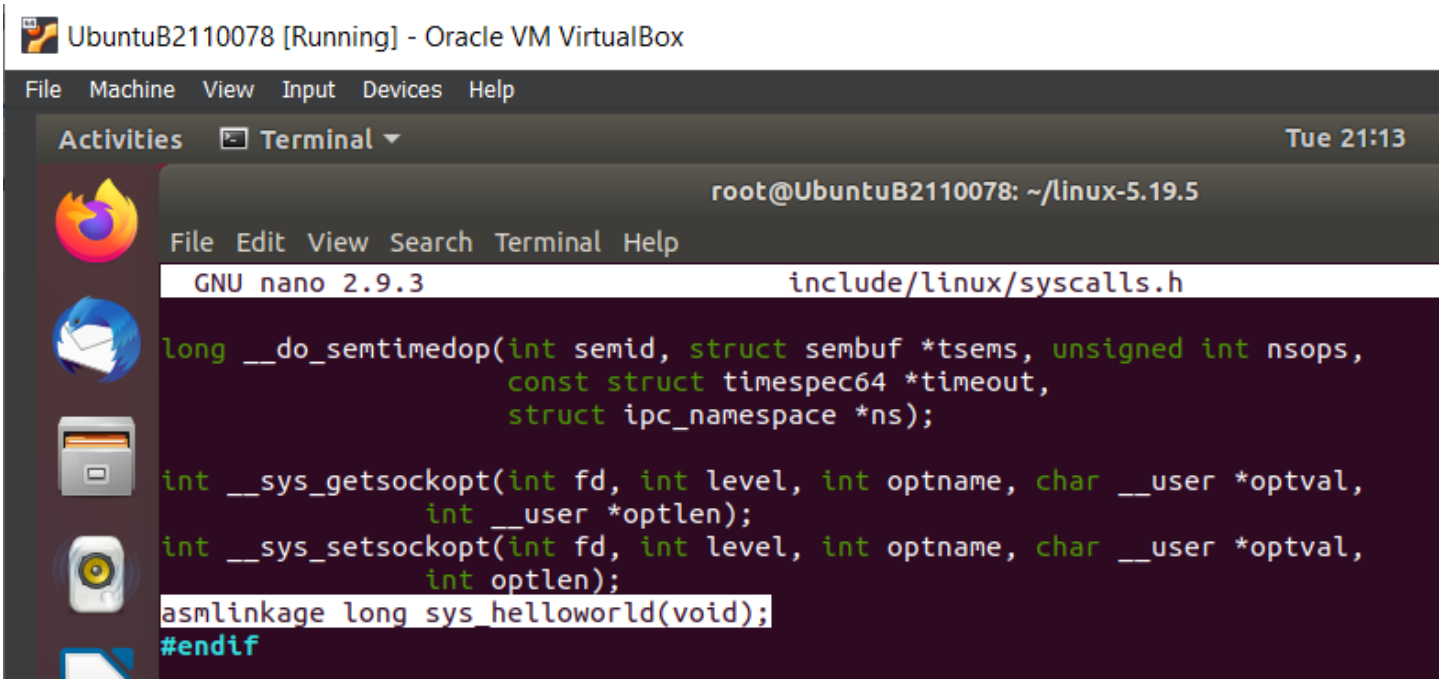
gedit Makefile

- Tìm dòng "**core-y += kernel/ certs/ mm/ fs/ ipc/ security/ crypto/**"
- Sửa thành "**core-y += kernel/ certs/ mm/ fs/ ipc/ security/ crypto/ helloworld/**"

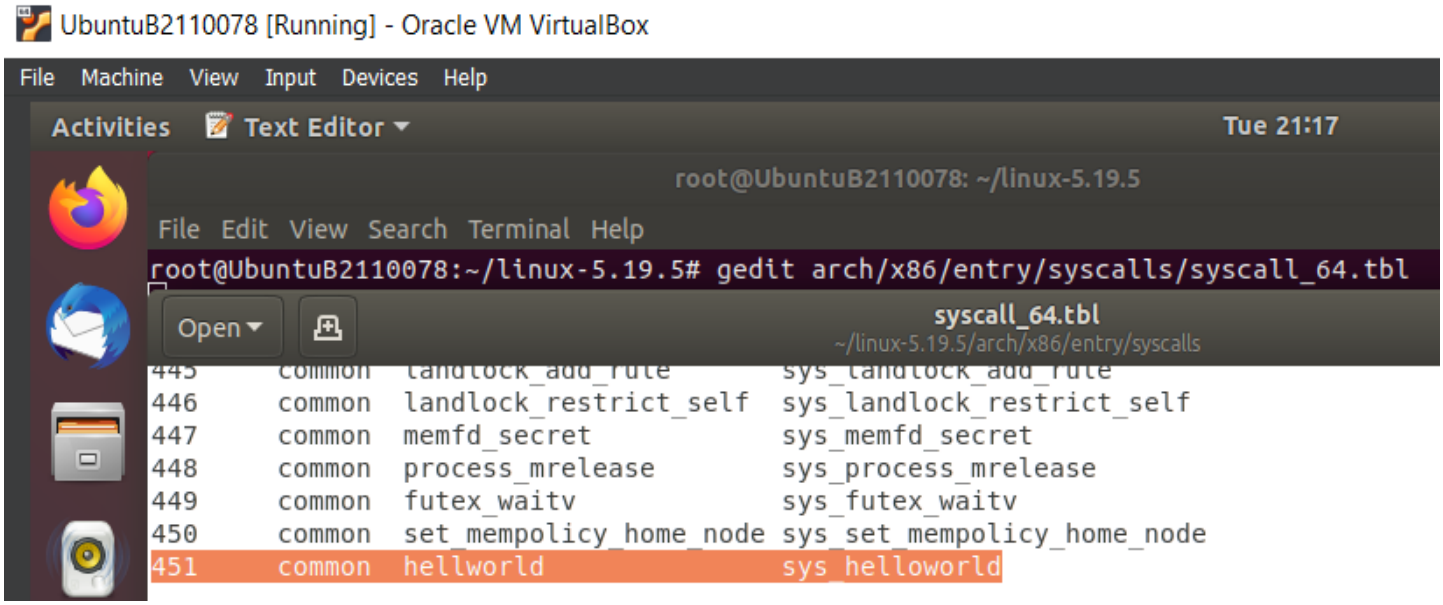


- Mở file include/linux/syscalls.h để chèn lời gọi hệ thống mới vào files syscalls header: **asmlinkage long sys_helloworld(void);**





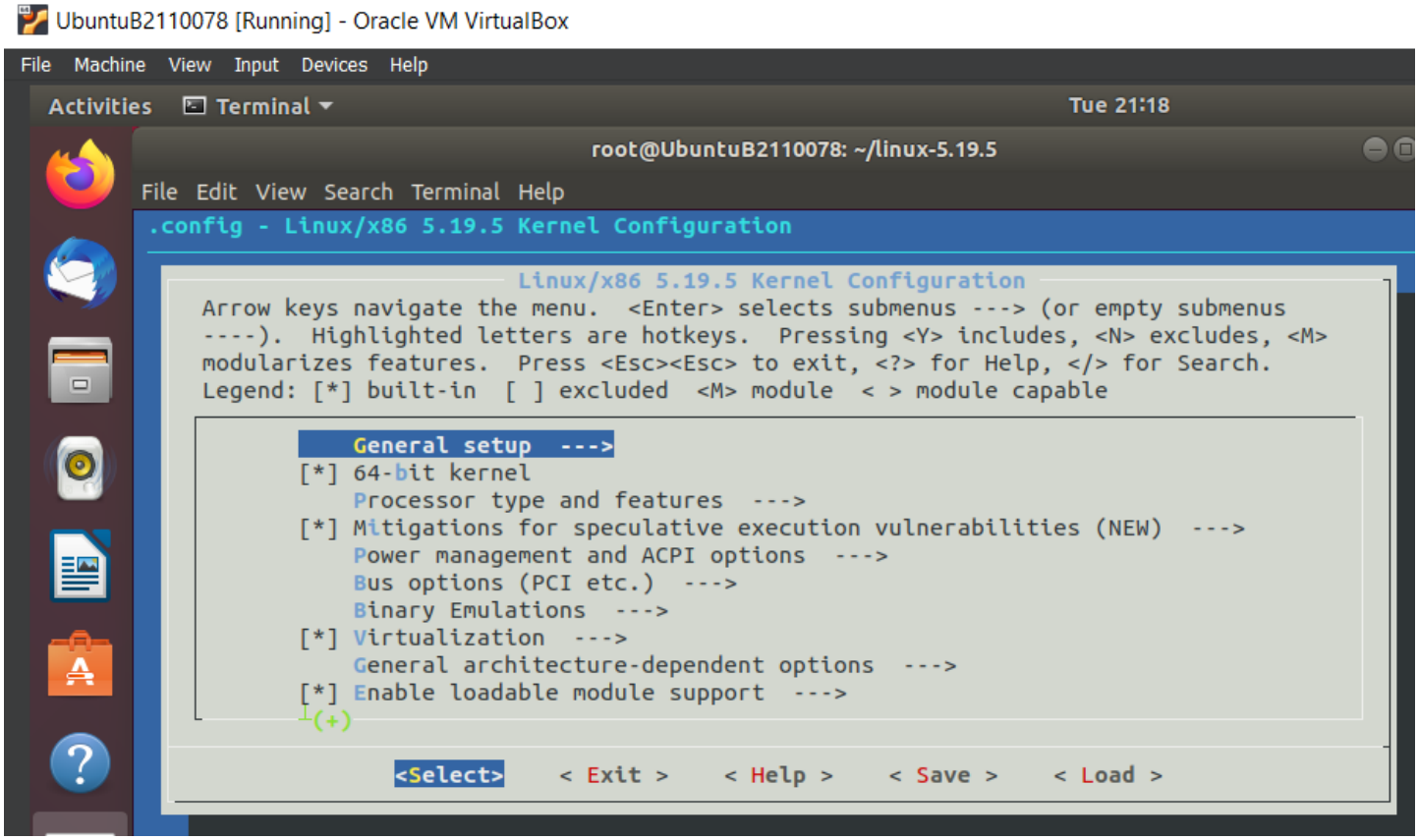
- Mở file arch/x86/entry/syscalls/syscall_64.tbl
- Chèn lời gọi hệ thống mới vào syscalls table:
  451 common helloworld sys_helloworld



C. THIẾT LẬP CẤU HÌNH KERNEL MỚI

- Tạo file cấu hình

 # make menuconfig

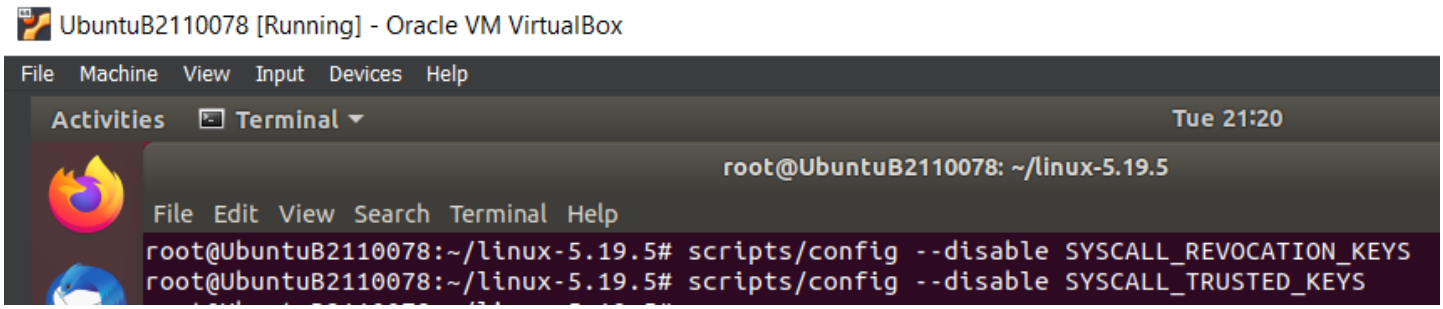


- Chạy lệnh vô hiệu hóa tính năng thu hồi lời gọi hệ thống

 # scripts/config –disable – SYSCALL_REVOCATION_KEYS

- Chạy lệnh vô hiệu hóa tính năng xác thực chìa khóa

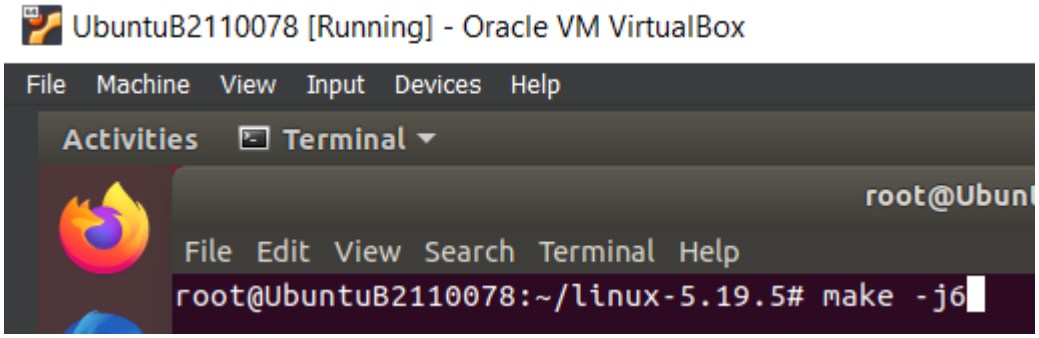
 # scripts/config –disable – SYSCALL_TRUSTED_KEYS

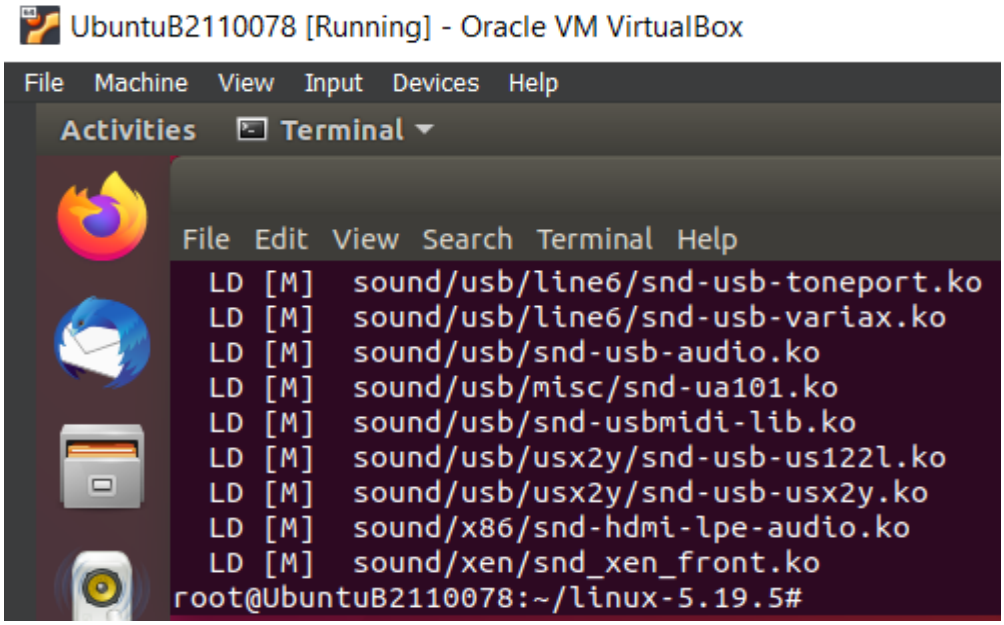


D. BIÊN DỊCH KERNEL

- Tại **~/linux-5.19.5**, tạo kernel image nén

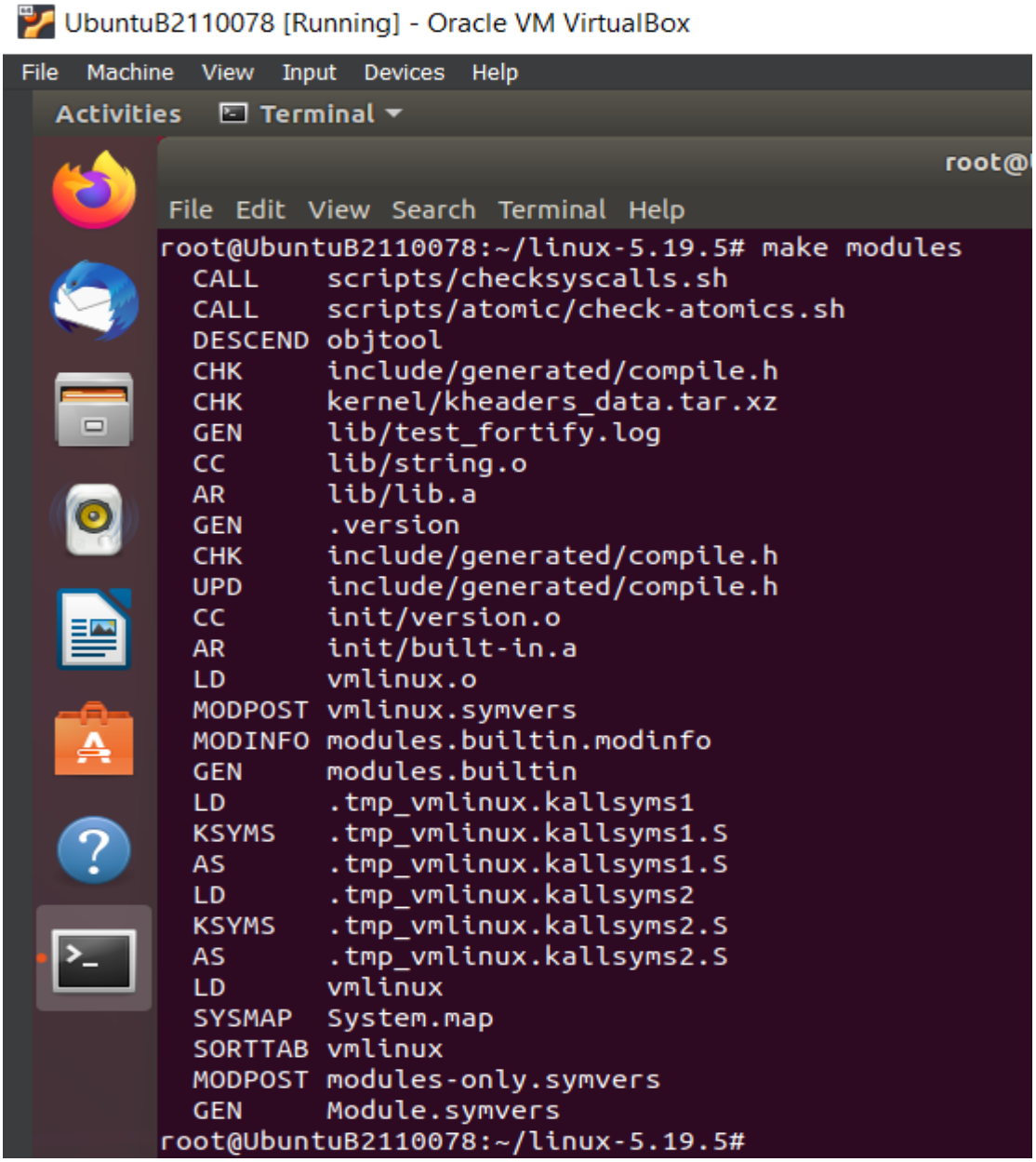
make -j6





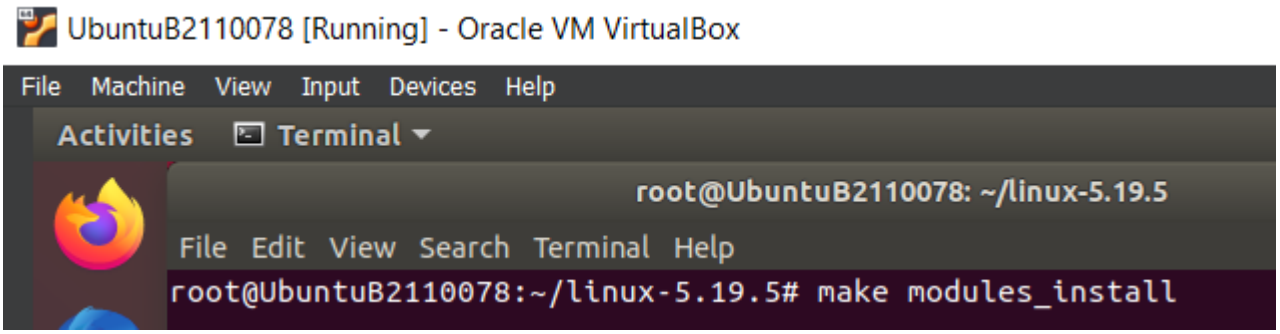
- Biên dịch kernel modules

make modules



- Cài đặt kernel modules

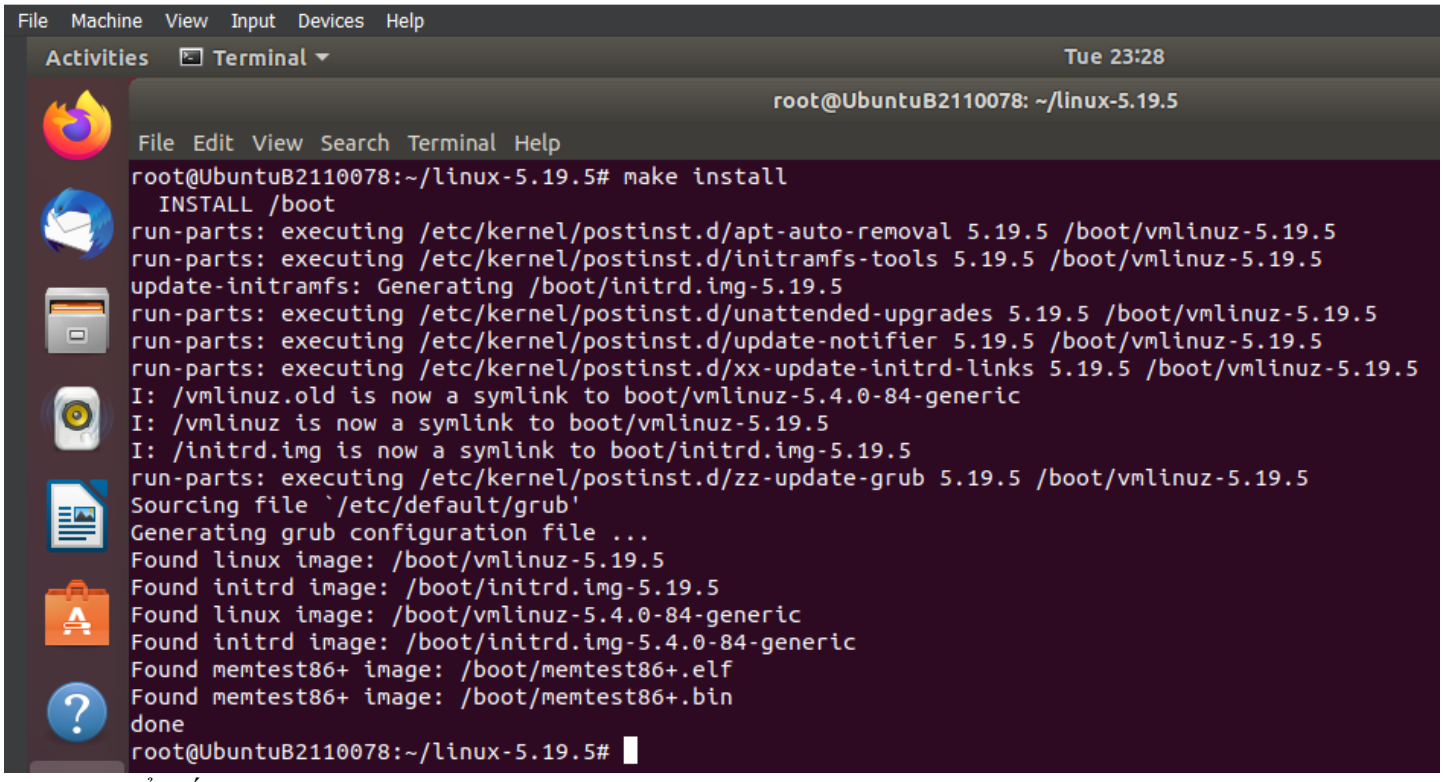
make modules_install



- Cài đặt kernel

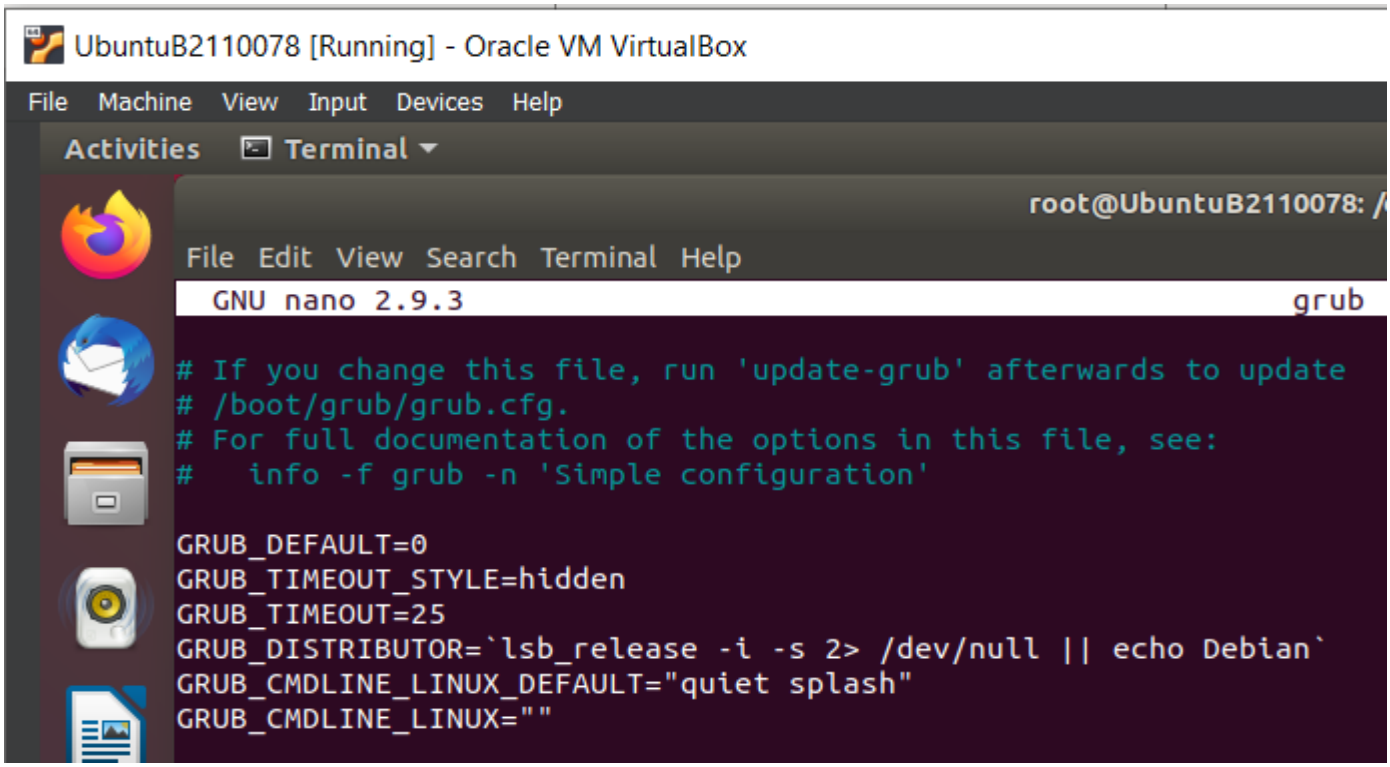
make install

UbuntuB2110078 [Running] - Oracle VM VirtualBox



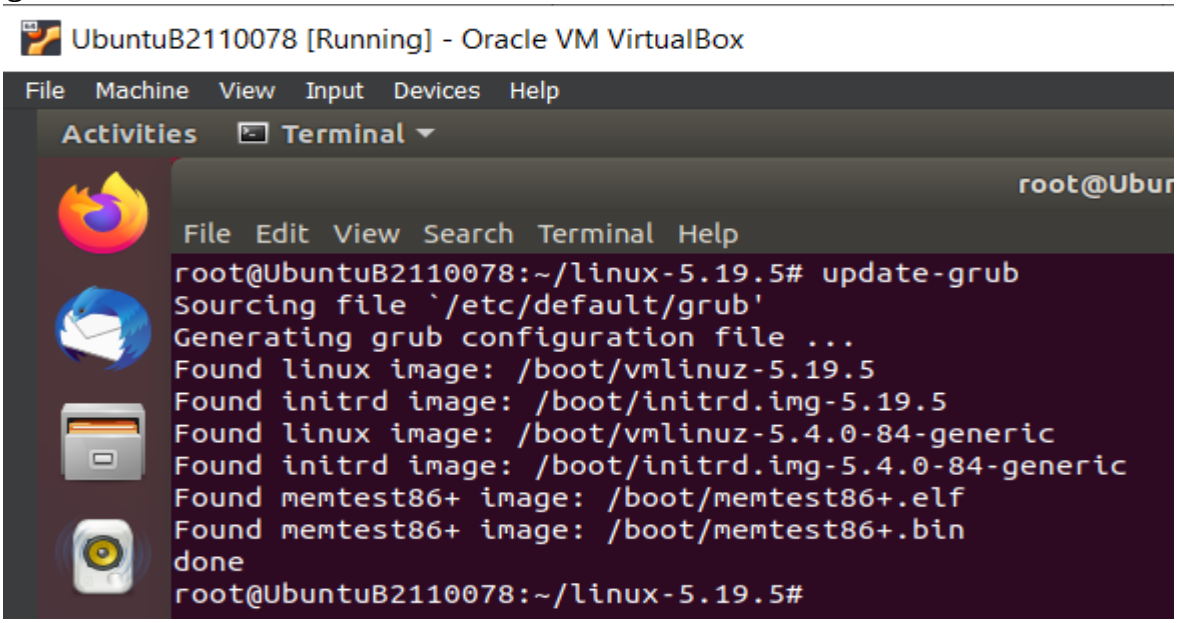
- Thay đổi cấu hình grub:

GRUB_DEFAULT=0
GRUB_TIMEOUT=25



- Cập nhật cấu hình grub

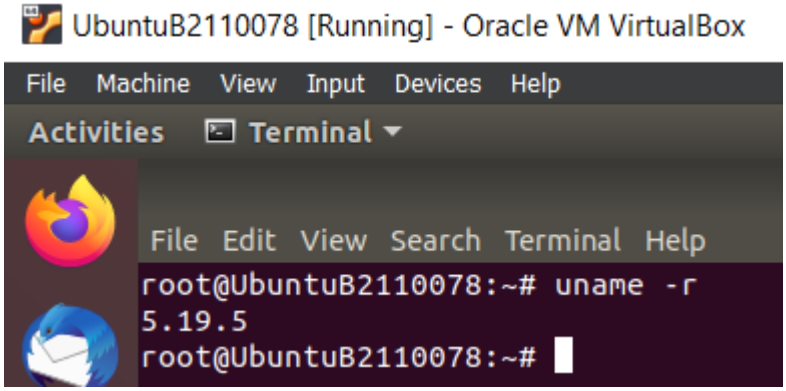
update-grub



- Reboot kernel mới và kiểm tra phiên bản kernel

reboot

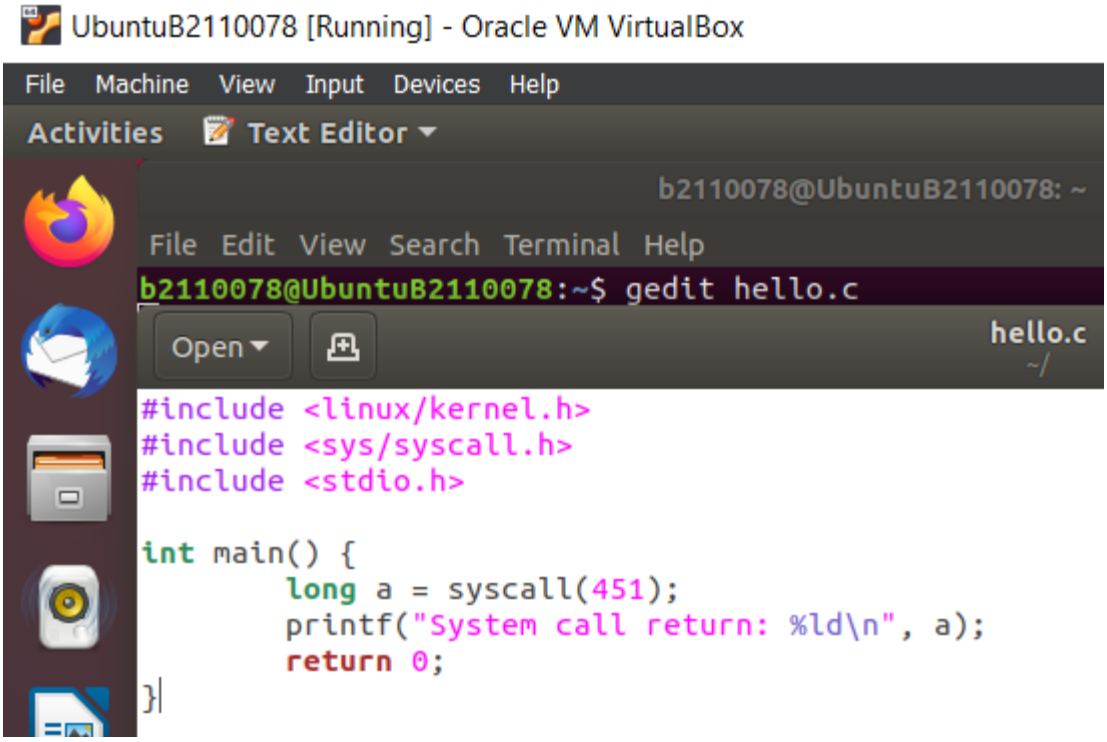
uname -r



E. THỬ NGHIỆM SYSCALL MỚI

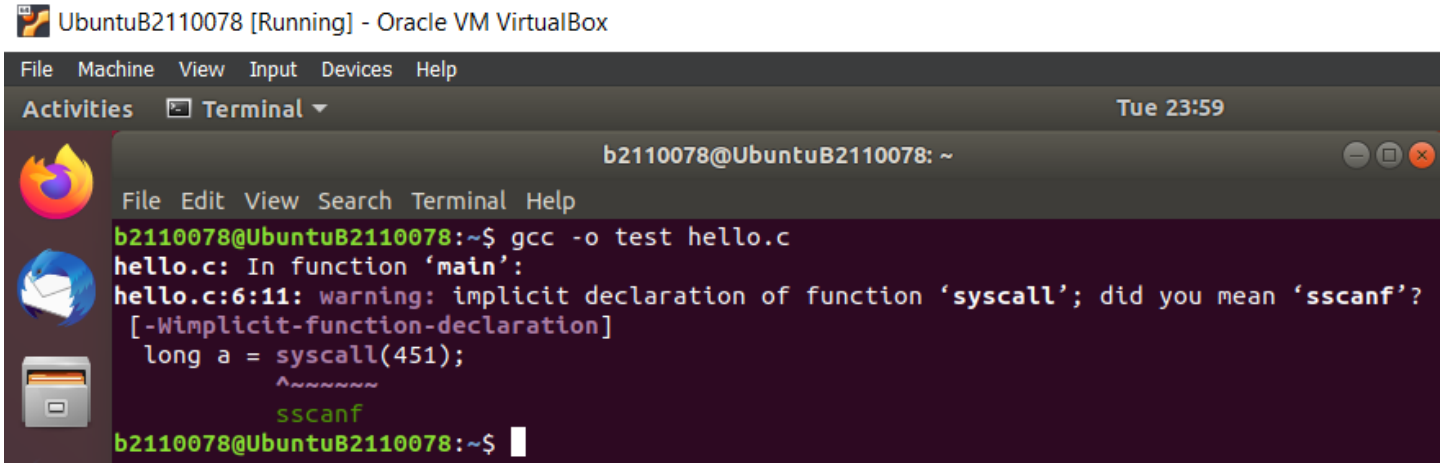
- Tạo file hello.c với đoạn code bên dưới:

```
#include <linux/kernel.h>
#include <sys/syscall.h>
#include <stdio.h>

int main() {
        long a = syscall(451);
        printf("System call return: %ld\n", a);
        return 0;
}
```



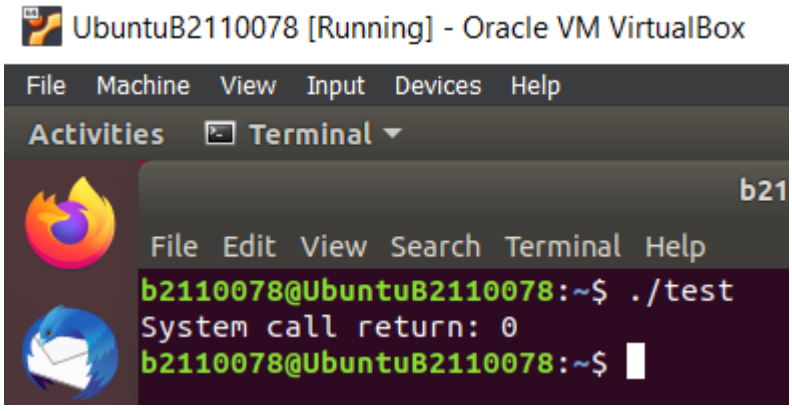
- Biên dịch chương trình hello.c và tạo file output test

gcc -o test hello.c



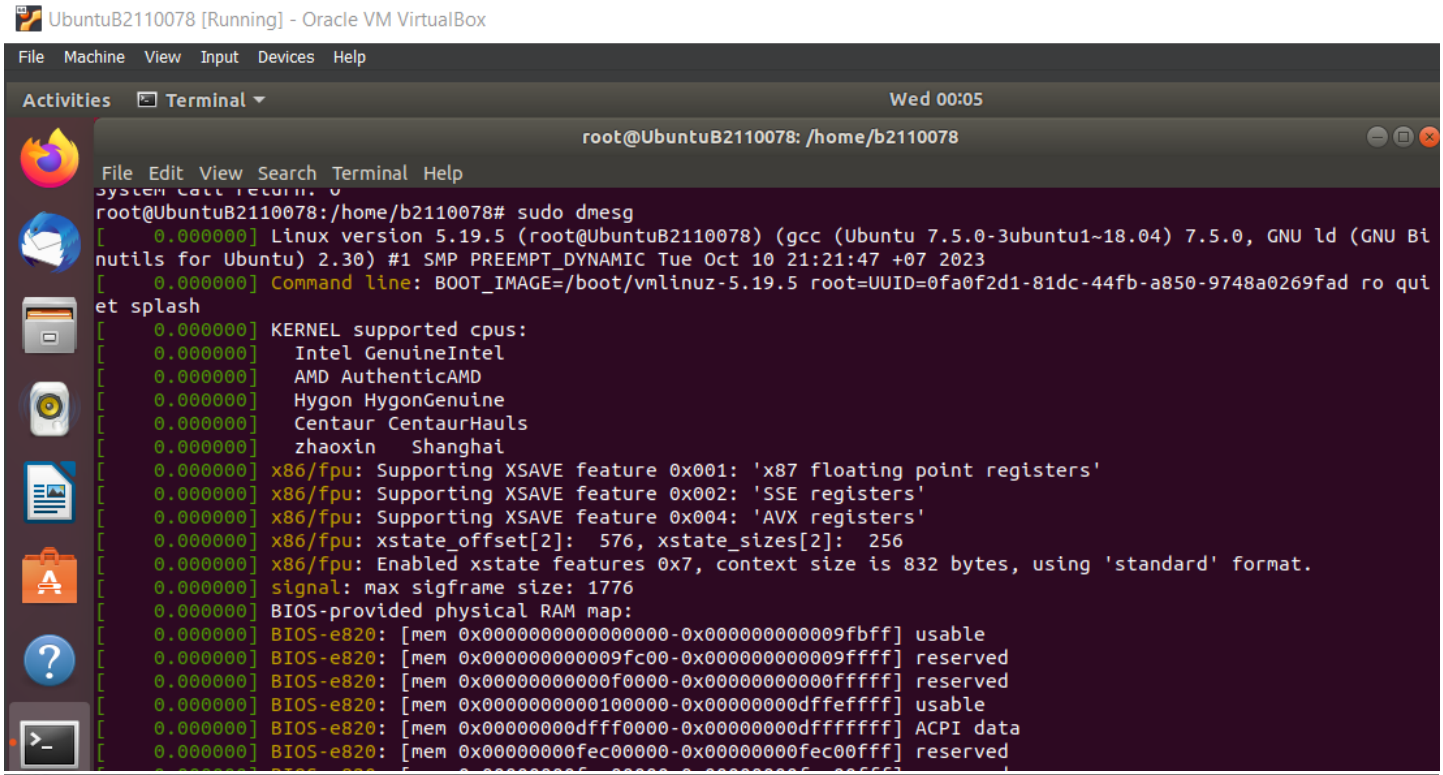
- Thực hiện file test

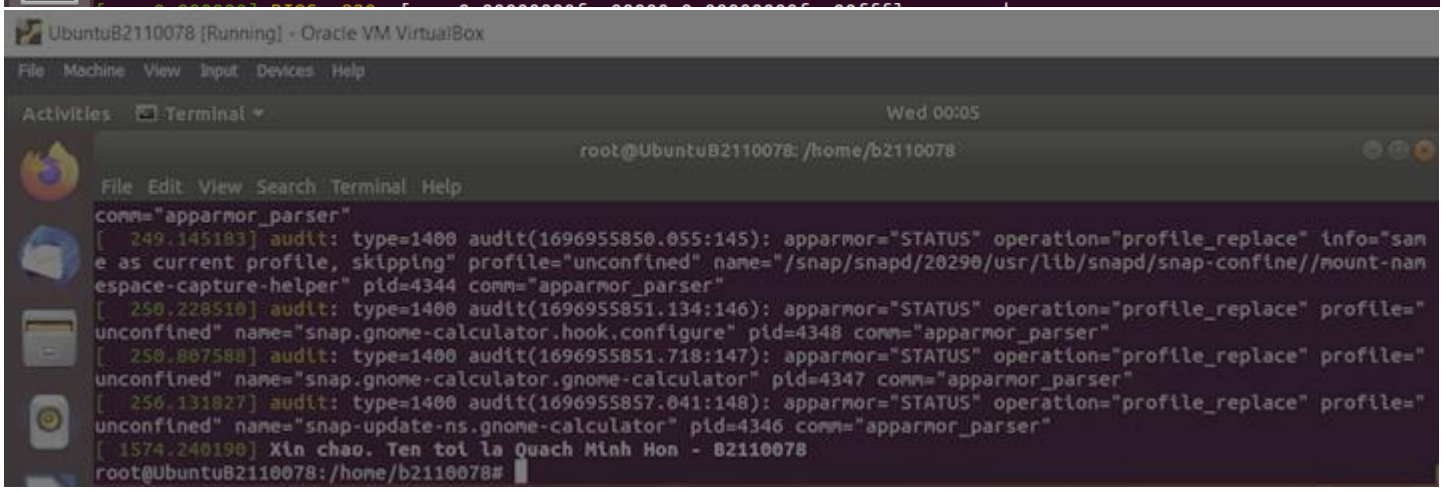
./test



- Nếu giá trị trở về là 0 thì lời gọi hệ thống đã được thực thi thành công.
- Khi xem log của kernel, ta có thể thấy câu "Xin chao. Ten toi la Quach Minh Hon – B2110078"

sudo dmesg





PHẦN 3: Cấp phát vùng nhớ

Anh chị hãy cài đặt 3 thuật toán cấp phát vùng nhớ (First-fit, Best-Fit, Worst-Fit) cho tiến trình, đưa ví dụ cụ thể để test kết quả thực hiện.

- **Thuật toán First-fit:**

UbuntuB2110078 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Visual Studio Code Oct 12 14:53

First_Fit.c - Partitions - Visual Studio Code

File Edit Selection View Go Run Terminal Help

First_Fit.c

First_Fit.c > First_Fit(int [], int [], int, int)

```c
#include <stdio.h>

void First_Fit(int phanVung[], int tienTrinh[], int parSize, int proSize) {
    int i, j, check[proSize], tmp[parSize];

    for (i=0; i<parSize; i++)
        tmp[i] = phanVung[i];
    // gan cac phan tu cua mang phanVung vao mang tmp
    // cap nhat lai mang tmp sau moi lan duyet
    // mang tmp chi dung de printf

    for (i=0; i<proSize; i++) {
        check[i] = 0; // gan gia tri 0 de tim tien trinh nao chua duoc allocated
        printf("%d: ", i+1); // print thu tu cua moi lan duyet

        for (j=0; j<parSize; j++) {
            if (tienTrinh[i] <= phanVung[j]) {
                phanVung[j] -= tienTrinh[i]; // lay phanVung - tienTrinh
                printf("%d --%d--> %d => %d is allocated to partition No.%d", tmp[j], tienTrinh[i],
                phanVung[j], tienTrinh[i], j+1);
                check[i] = 1; // neu tienTrinh duoc duyet thi gan check = 1
                break;        // va break khoi vong lap
            }
        }
        // neu tienTrinh[i] van bang 0 sau khi duyet thi thong bao chua duoc allocated
        if (!check[i])
            printf("%d is not allocated", tienTrinh[i]);
        printf("\n");
    }
}
```

UbuntuB2110078 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Visual Studio Code Oct 12 14:46

First_Fit.c - Partitions - Visual Studio Code

File Edit Selection View Go Run Terminal Help

First_Fit.c

First_Fit.c > ...

```c
void printPartitionID(int phanVung[], int parSize) {
    int i;
    printf("|-----|-------------|\n");
    printf("| No. |\t Paritions  |\n");
    printf("|-----|-------------|\n");
    for (i=0; i<parSize; i++)
        printf("|  %d  |\t   %d     |\n", i+1, phanVung[i]);
    printf("|-----|-------------|\n");
    printf("\n");
}

void printParAfter(int phanVung[], int parSize) {
    int i;
    printf("\nCac kich thuoc phan vung sau khi allocate:\n");
    for (i=0; i<parSize; i++)
        printf("%d ", phanVung[i]);
    printf("\n");
}

int main() {
    int phanVung[] = {300, 600, 350, 200, 750, 125};
    int tienTrinh[] = {115, 500, 358, 200, 375, 1300};
    int parSize = sizeof(phanVung) / sizeof(phanVung[0]); // Lay size cua mang phanVung
    int proSize = sizeof(tienTrinh) / sizeof(tienTrinh[0]); // Lay size cua mang tienTrinh

    printPartitionID(phanVung, parSize); // print ID cua cac phan vung
    First_Fit(phanVung, tienTrinh, parSize, proSize);
    printParAfter(phanVung, parSize);
}
```

UbuntuB2110078 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Terminal

b2110078@UbuntuB2110078: ~/P

```
b2110078@UbuntuB2110078:~/Partitions$ gcc -o firstFit First_Fit.c
b2110078@UbuntuB2110078:~/Partitions$ ./firstFit
|-----|-------------|
| No. |  Paritions  |
|-----|-------------|
|  1  |     300     |
|  2  |     600     |
|  3  |     350     |
|  4  |     200     |
|  5  |     750     |
|  6  |     125     |
|-----|-------------|

1: 300 --115--> 185 => 115 is allocated to partition No.1
2: 600 --500--> 100 => 500 is allocated to partition No.2
3: 750 --358--> 392 => 358 is allocated to partition No.5
4: 350 --200--> 150 => 200 is allocated to partition No.3
5: 750 --375--> 17 => 375 is allocated to partition No.5
6: 1300 is not allocated

Cac kich thuoc phan vung sau khi allocate:
185 100 150 200 17 125
b2110078@UbuntuB2110078:~/Partitions$
```

- **Thuật toán Best-fit**

UbuntuB2110078 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Visual Studio Code Oct 12 15

Best_Fit.c - Partitions - V

File Edit Selection View Go Run Terminal Help

First_Fit.c | Best_Fit.c ×

Best_Fit.c > minimum(int [], int)

```c
#include <stdio.h>
#include <limits.h>

int minimum(int bestFit[], int parSize) {
    int i, min=INT_MAX;
    // gan min = gia tri max cua integer
    for (i=1; i<parSize; i++)
        if (bestFit[i] < min && bestFit[i] != -1)
            min = bestFit[i];
    // tim gia tri nho nhat nhung lon hon -1
    // neu khong tim duoc gia tri nho nhat thi tra ve INT_MAX
    return min;
}
```

UbuntuB2110078 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Visual Studio Code Oct 12 15:06

Best_Fit.c - Partitions - Visual Studio Code

File Edit Selection View Go Run Terminal Help

First_Fit.c | Best_Fit.c

Best_Fit.c > minimum(int [], int)

```c
void Best_Fit(int phanVung[], int tienTrinh[], int parSize, int proSize) {
    int i, j, parID[6], check[6], tmp[6];
    int bestFit[6], id, min;

    for (j=0; j<parSize; j++)
        tmp[j] = phanVung[j];
    // gan cac phan tu cua mang phanVung vao mang tmp
    // cap nhat lai mang tmp sau moi lan duyet
    // mang tmp chi dung de printf

    for (i=0; i<proSize; i++) {
        check[i] = 0; // gan gia tri 0 de tim tien trinh nao chua duoc allocated
        printf("%d: ", i+1); // print thu tu cua moi lan duyet

        for (j=0; j<parSize; j++)
            bestFit[j] = -1;
        // reset lai mang bestFit
        // gan gia tri -1 vao tat ca cac phan tu cua mang

        for (j=0; j<parSize; j++)
            if (tienTrinh[i] <= phanVung[j])
                bestFit[j] = phanVung[j];
        // gan lai cac gia tri cua phanVung >= tienTrinh vao mang bestFit

        min = minimum(bestFit, parSize);
        for (j=0; j<parSize; j++) {
            if (tienTrinh[i] <= phanVung[j] && phanVung[j] == min) {
                phanVung[j] -= tienTrinh[i];
                printf("%d --%d--> %d => %d is allocated to partition No.%d\n", tmp[j], tienTrinh[i],
                phanVung[j], tienTrinh[i], j+1);
                check[i] = 1; // neu tienTrinh duoc duyet thi gan check = 1
                break;          // va break khoi vong lap
```

UbuntuB2110078 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Visual Studio Code Oct 12 1

Best_Fit.c - Partitions -

File Edit Selection View Go Run Terminal Help

First_Fit.c Best_Fit.c

Best_Fit.c > ...

```
46                  break;          // va break khoi vong lap
47              }
48          }
49
50          if (!check[i])
51              printf("%d is not allocated\n", tienTrinh[i]);
52      }
53  }
54
55  void printPartitionID(int phanVung[], int parSize) {
56      int i;
57      printf("|-----|-------------|\n");
58      printf("| No. |  Paritions  |\n");
59      printf("|-----|-------------|\n");
60      for (i=0; i<parSize; i++)
61          printf("|  %d  |     %d      |\n", i+1, phanVung[i]);
62      printf("|-----|-------------|\n");
63      printf("\n");
64  }
65
66  void printParAfter(int phanVung[], int parSize) {
67      int i;
68      printf("\nCac kich thuoc phan vung sau khi allocate:\n");
69      for (i=0; i<parSize; i++)
70          printf("%d ", phanVung[i]);
71      printf("\n");
72  }
```

UbuntuB2110078 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Visual Studio Code Oct 12 15:09

Best_Fit.c - Partitions - Visual Studio Code

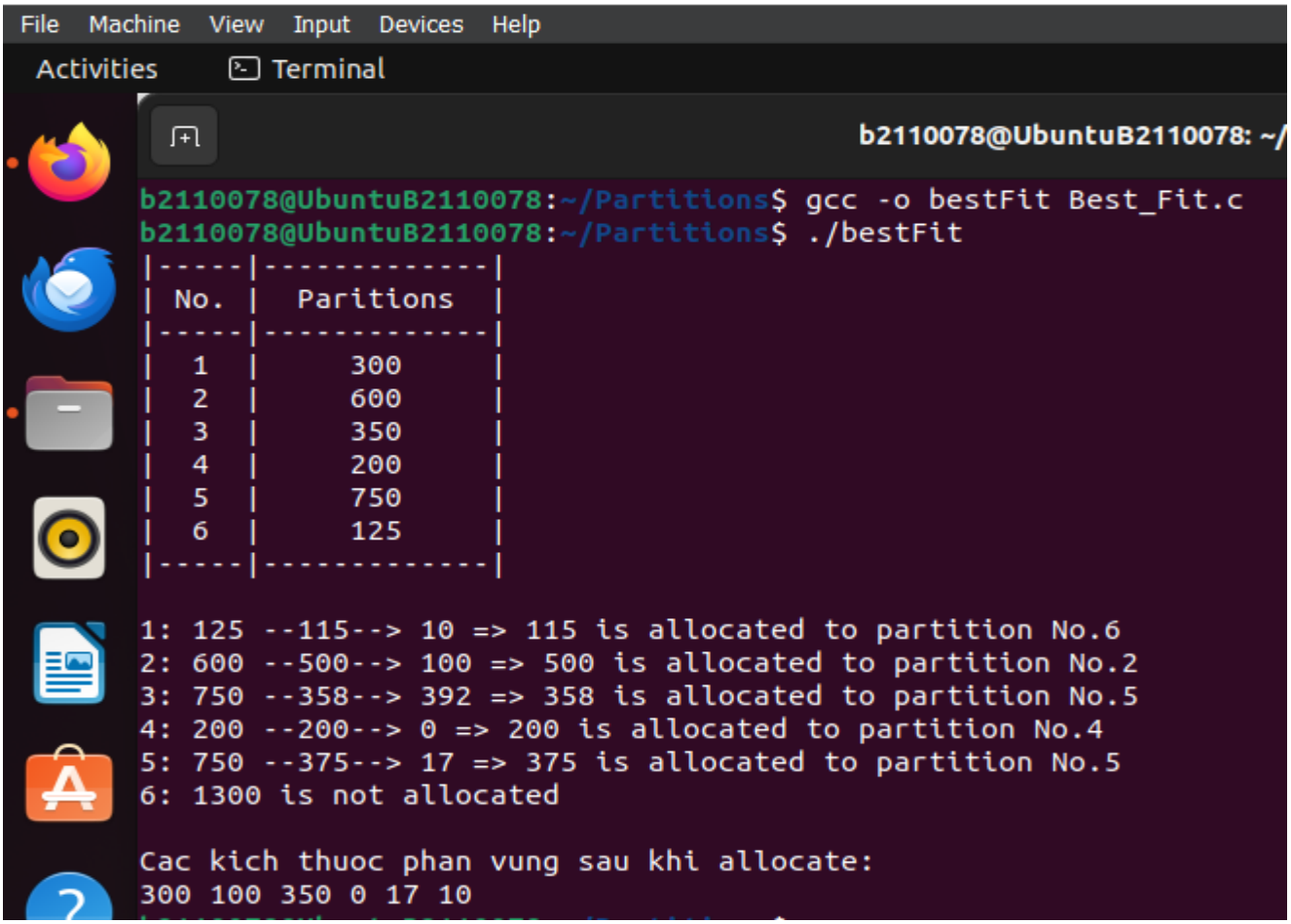
File Edit Selection View Go Run Terminal Help

First_Fit.c Best_Fit.c

Best_Fit.c > ...

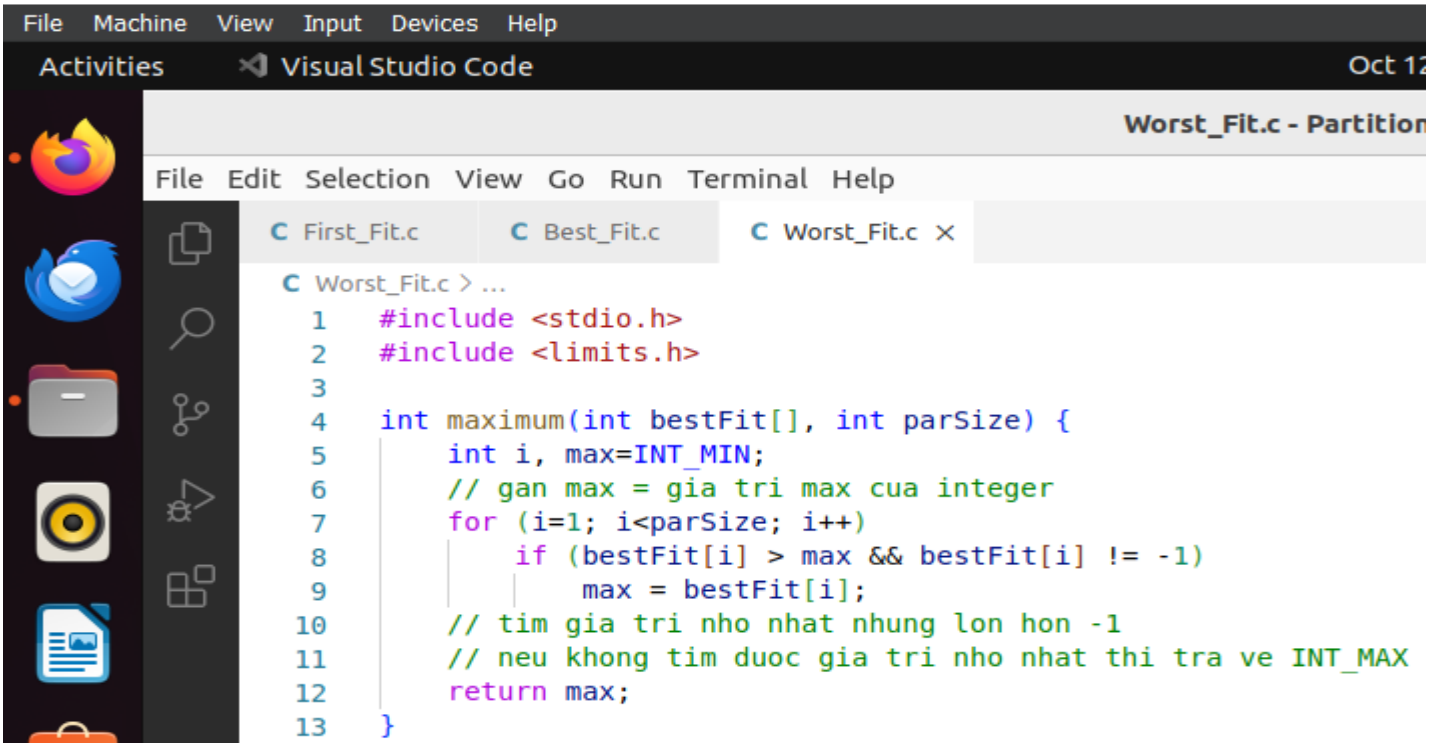
```
73
74  int main() {
75      int phanVung[] = {300, 600, 350, 200, 750, 125};
76      int tienTrinh[] = {115, 500, 358, 200, 375, 1300};
77      int parSize = sizeof(phanVung) / sizeof(phanVung[0]); // Lay size cua mang phanVung
78      int proSize = sizeof(tienTrinh) / sizeof(tienTrinh[0]); // Lay size cua mang tienTrinh
79
80      printPartitionID(phanVung, parSize); // print ID cua cac phan vung
81      Best_Fit(phanVung, tienTrinh, parSize, proSize);
82      printParAfter(phanVung, parSize); // print kich thuoc cac phan vung sau khi cap phat
83  }
```

UbuntuB2110078 [Running] - Oracle VM VirtualBox



- **Thuật toán Worst-fit**

UbuntuB2110078 [Running] - Oracle VM VirtualBox



UbuntuB2110078 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Activities Visual Studio Code Oct 12 15:22

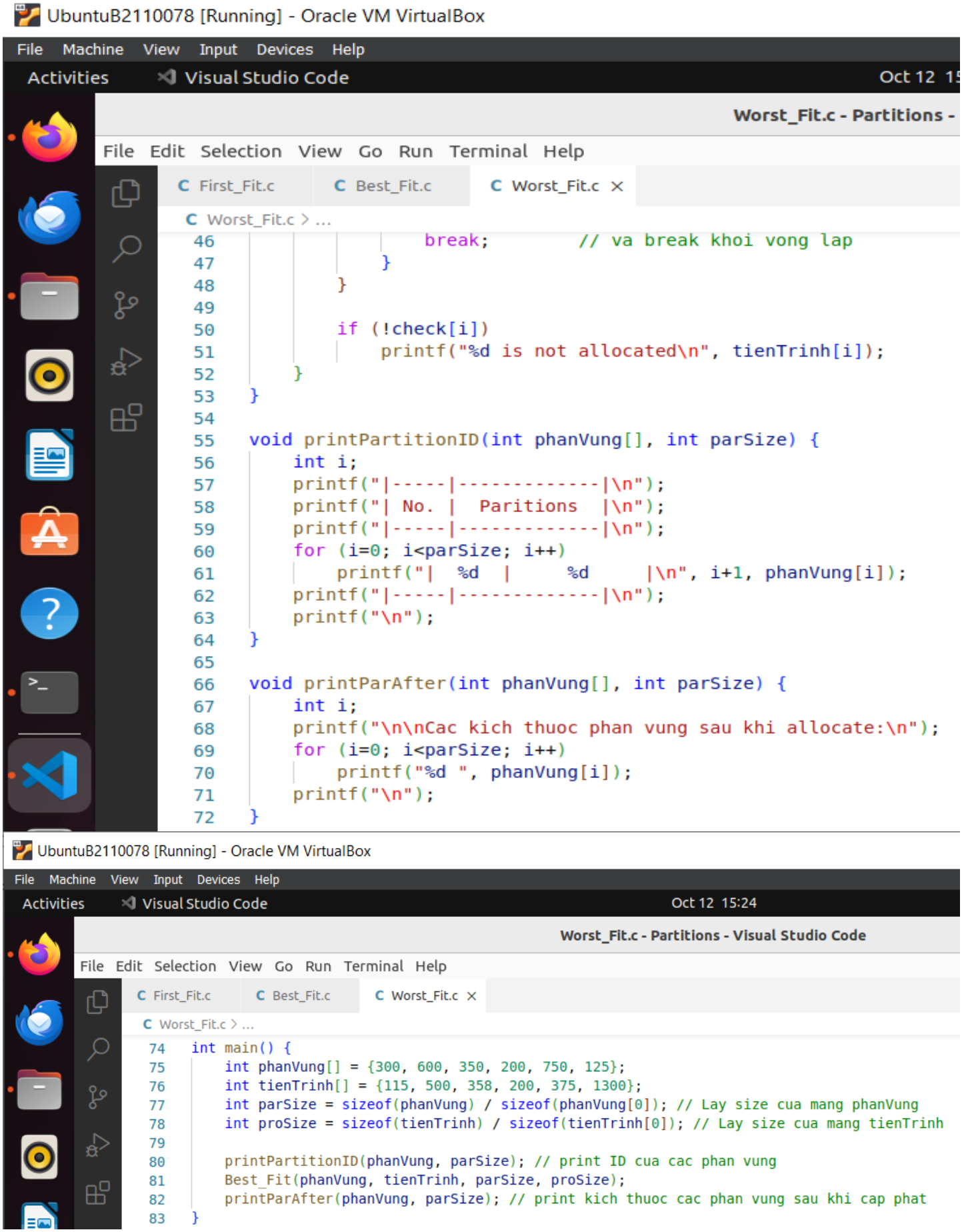
Worst_Fit.c - Partitions - Visual Studio Code

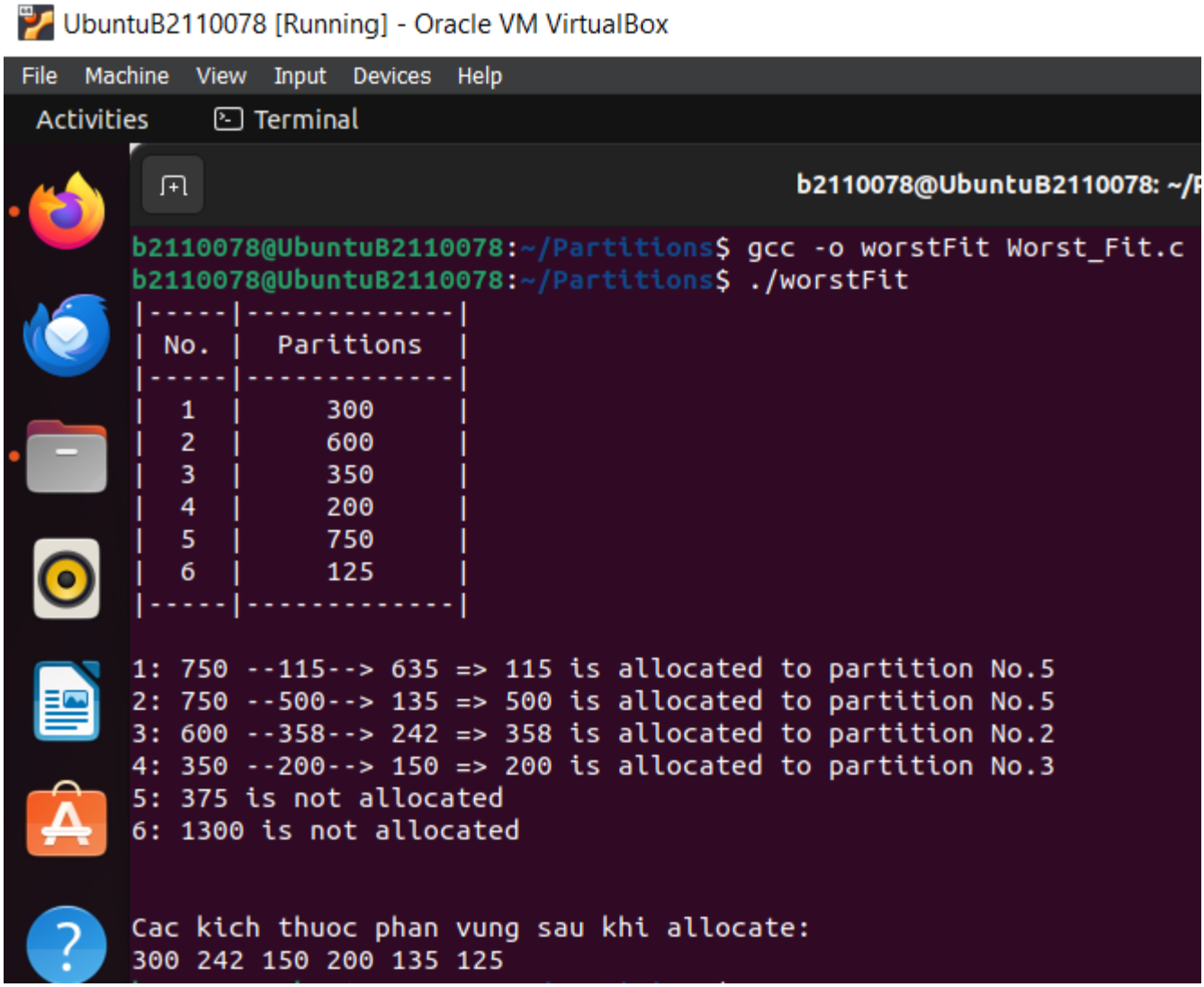
File Edit Selection View Go Run Terminal Help

First_Fit.c | Best_Fit.c | Worst_Fit.c

Worst_Fit.c > ...

```c
void Best_Fit(int phanVung[], int tienTrinh[], int parSize, int proSize) {
    int i, j, parID[6], check[6], tmp[6];
    int bestFit[6], id, max;

    for (j=0; j<parSize; j++)
        tmp[j] = phanVung[j];
    // gan cac phan tu cua mang phanVung vao mang tmp
    // cap nhat lai mang tmp sau moi lan duyet
    // mang tmp chi dung de printf

    for (i=0; i<proSize; i++) {
        check[i] = 0; // gan gia tri 0 de tim tien trinh nao chua duoc allocated
        printf("%d: ", i+1); // print thu tu cua moi lan duyet

        for (j=0; j<parSize; j++)
            bestFit[j] = -1;
        // reset lai mang bestFit
        // gan gia tri -1 vao tat ca cac phan tu cua mang

        for (j=0; j<parSize; j++)
            if (tienTrinh[i] <= phanVung[j])
                bestFit[j] = phanVung[j];
        // gan lai cac gia tri cua phanVung >= tienTrinh vao mang bestFit

        max = maximum(bestFit, parSize);
        for (j=0; j<parSize; j++) {
            if (tienTrinh[i] <= phanVung[j] && phanVung[j] == max) {
                phanVung[j] -= tienTrinh[i];
                printf("%d --%d--> %d => %d is allocated to partition No.%d\n", tmp[j], tienTrinh[i],
                phanVung[j], tienTrinh[i], j+1);
                check[i] = 1; // neu tienTrinh duoc duyet thi gan check = 1
                break;        // va break khoi vong lap
```


UbuntuB2110078 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Visual Studio Code
Oct 12
Worst_Fit.c - Partitions -
File Edit Selection View Go Run Terminal Help
First_Fit.c
Best_Fit.c
Worst_Fit.c
Worst_Fit.c > ...
46 break; // va break khoi vong lap
47 }
48 }
49
50 if (!check[i])
51 printf("%d is not allocated\n", tienTrinh[i]);
52 }
53 }
54
55 void printPartitionID(int phanVung[], int parSize) {
56 int i;
57 printf("|-----|-------------|\n");
58 printf("| No. | Paritions |\n");
59 printf("|-----|-------------|\n");
60 for (i=0; i<parSize; i++)
61 printf("| %d | %d |\n", i+1, phanVung[i]);
62 printf("|-----|-------------|\n");
63 printf("\n");
64 }
65
66 void printParAfter(int phanVung[], int parSize) {
67 int i;
68 printf("\n\nCac kich thuoc phan vung sau khi allocate:\n");
69 for (i=0; i<parSize; i++)
70 printf("%d ", phanVung[i]);
71 printf("\n");
72 }
UbuntuB2110078 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Visual Studio Code
Oct 12 15:24
Worst_Fit.c - Partitions - Visual Studio Code
File Edit Selection View Go Run Terminal Help
First_Fit.c
Best_Fit.c
Worst_Fit.c
Worst_Fit.c > ...
74 int main() {
75 int phanVung[] = {300, 600, 350, 200, 750, 125};
76 int tienTrinh[] = {115, 500, 358, 200, 375, 1300};
77 int parSize = sizeof(phanVung) / sizeof(phanVung[0]); // Lay size cua mang phanVung
78 int proSize = sizeof(tienTrinh) / sizeof(tienTrinh[0]); // Lay size cua mang tienTrinh
79
80 printPartitionID(phanVung, parSize); // print ID cua cac phan vung
81 Best_Fit(phanVung, tienTrinh, parSize, proSize);
82 printParAfter(phanVung, parSize); // print kich thuoc cac phan vung sau khi cap phat
83 }


UbuntuB2110078 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Terminal
b2110078@UbuntuB2110078: ~/P
b2110078@UbuntuB2110078:~/Partitions$ gcc -o worstFit Worst_Fit.c
b2110078@UbuntuB2110078:~/Partitions$ ./worstFit
No.	Paritions
1	300
2	600
3	350
4	200
5	750
6	125
1: 750 --115--> 635 => 115 is allocated to partition No.5
2: 750 --500--> 135 => 500 is allocated to partition No.5
3: 600 --358--> 242 => 358 is allocated to partition No.2
4: 350 --200--> 150 => 200 is allocated to partition No.3
5: 375 is not allocated
6: 1300 is not allocated
Cac kich thuoc phan vung sau khi allocate:
300 242 150 200 135 125

**********HẾT**********