Project 1 Report

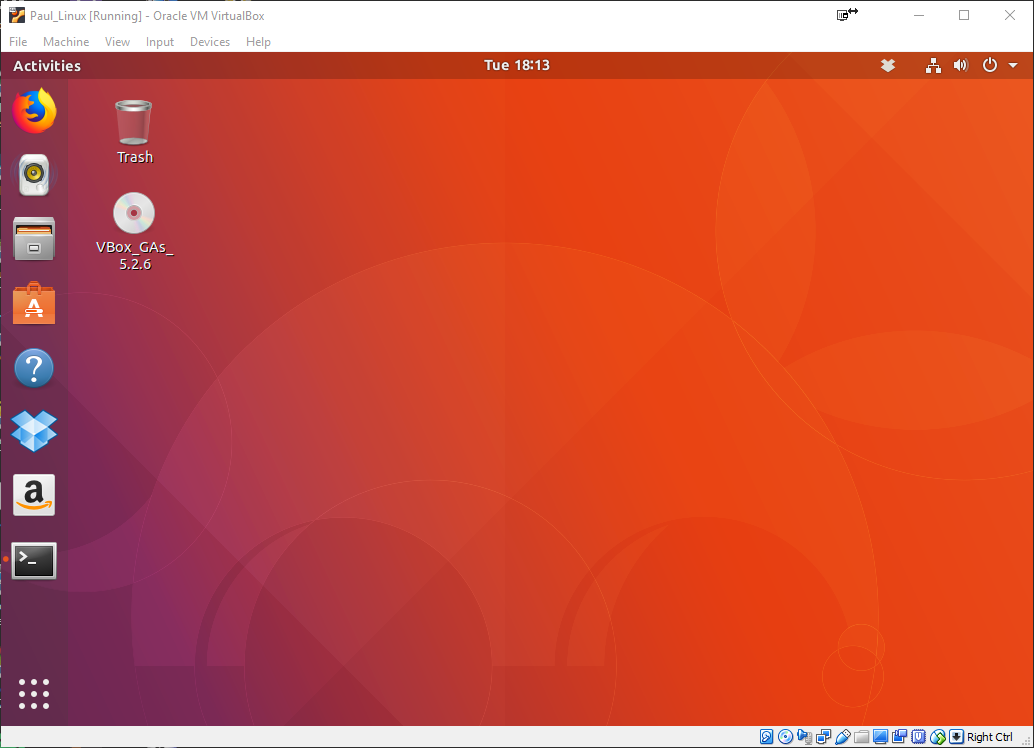
CSC 4320/6320 – Operating Systems

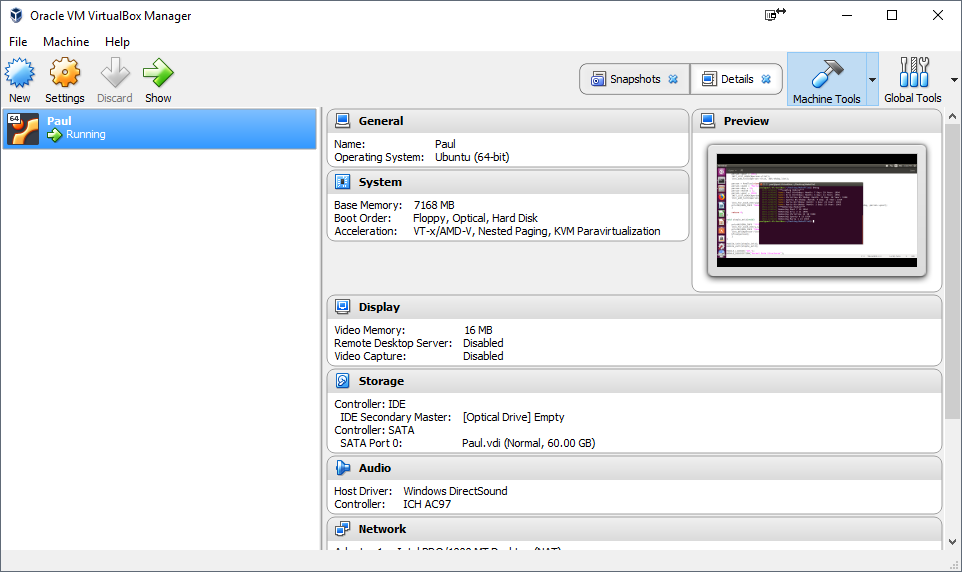
Spring 2018

Name: Paul Vlahos

Email: [pvlahos1@student.gsu.edu](mailto:pvlahos1@student.gsu.edu)

**Part 1:**





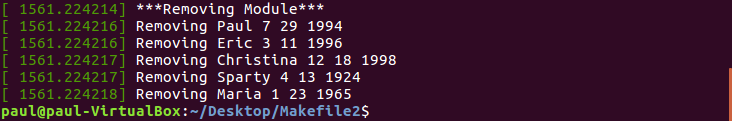
**Part 2:**





**Part 3:**





#include <linux/init.h>

#include <linux/module.h>

#include <linux/kernel.h>

#include <linux/list.h>

#include <linux/slab.h>

struct birthday

{

int month;

int day;

int year;

char \*name;

struct list\_head list;

};

/\*\*

\* The following defines and initializes a list\_head object named birthday\_list

\*/

static LIST\_HEAD(birthday\_list);

struct birthday \*person;

struct birthday \*next;

int simple\_init(void)

{

printk(KERN\_INFO "\*\*\*Loading Module\*\*\*\n");

person = kmalloc(sizeof(\*person), GFP\_KERNEL);

person->name = "Paul";

person->day = 29;

person->month = 7;

person->year = 1994;

INIT\_LIST\_HEAD(&person->list);

list\_add\_tail(&person->list, &birthday\_list);

person = kmalloc(sizeof(\*person), GFP\_KERNEL);

person->name = "Eric";

person->day = 11;

person->month = 3;

person->year = 1996;

INIT\_LIST\_HEAD(&person->list);

list\_add\_tail(&person->list, &birthday\_list);

person = kmalloc(sizeof(\*person), GFP\_KERNEL);

person->name = "Christina";

person->day = 18;

person->month = 12;

person->year = 1998;

INIT\_LIST\_HEAD(&person->list);

list\_add\_tail(&person->list, &birthday\_list);

person = kmalloc(sizeof(\*person), GFP\_KERNEL);

person->name = "Sparty";

person->day = 13;

person->month = 4;

person->year = 1924;

INIT\_LIST\_HEAD(&person->list);

list\_add\_tail(&person->list, &birthday\_list);

person = kmalloc(sizeof(\*person), GFP\_KERNEL);

person->name = "Maria";

person->day = 23;

person->month = 1;

person->year = 1965;

INIT\_LIST\_HEAD(&person->list);

list\_add\_tail(&person->list, &birthday\_list);

person = kmalloc(sizeof(\*person), GFP\_KERNEL);

person->name = "Maria";

person->day = 23;

person->month = 1;

person->year = 1965;

INIT\_LIST\_HEAD(&person->list);

list\_add\_tail(&person->list, &birthday\_list);

list\_for\_each\_entry(person, &birthday\_list, list) {

printk(KERN\_INFO "Name: %s Birthday: Month: %d Day: %d Year: %d", person->name, person->month, person->day, person->year);

}

return 0;

}

void simple\_exit(void)

{

printk(KERN\_INFO "\*\*\*Removing Module\*\*\*\n");

list\_for\_each\_entry\_safe(person,next,&birthday\_list,list) {

printk(KERN\_INFO "Removing %s %d %d %d", person->name, person->month, person->day, person->year);

list\_del(&person->list);

kfree(person);

}

}

module\_init(simple\_init);

module\_exit(simple\_exit);

MODULE\_LICENSE("GPL");

MODULE\_DESCRIPTION("Kernel Data Structures");

MODULE\_AUTHOR("SGG");