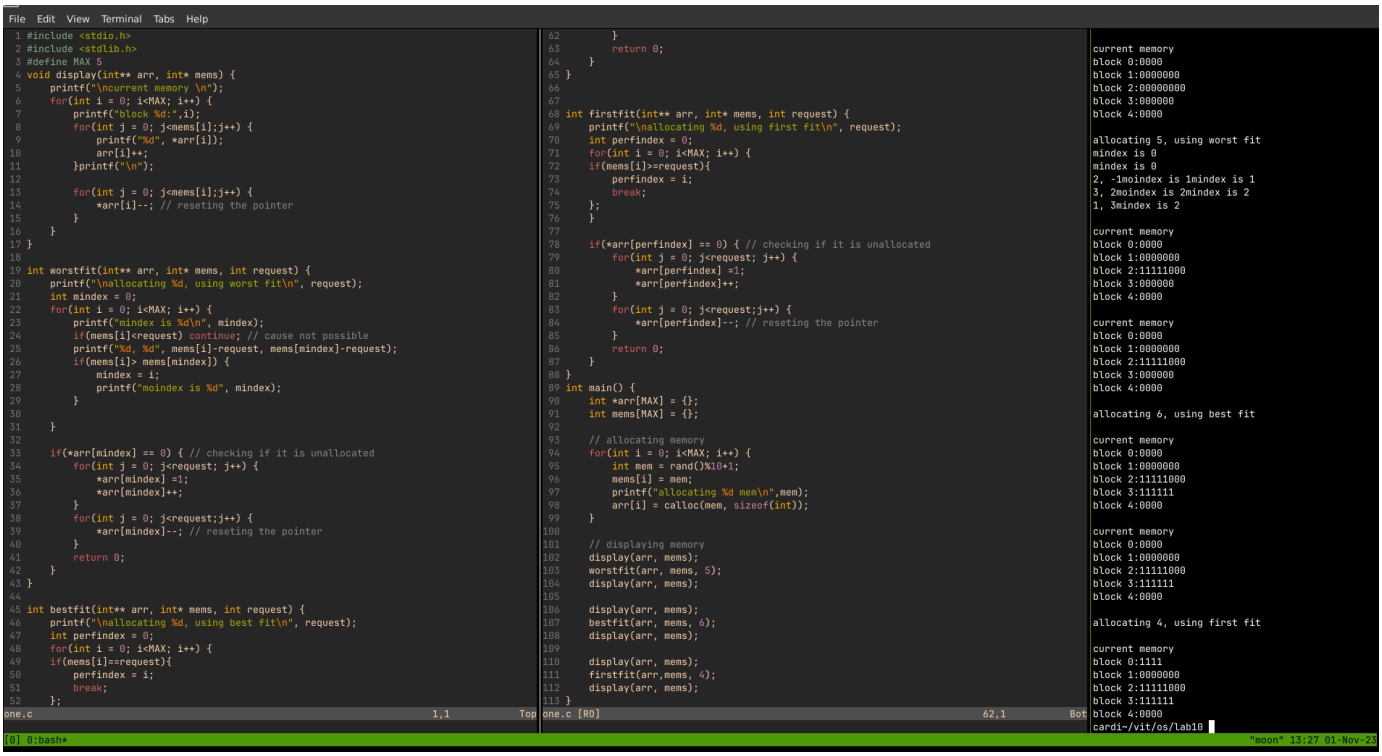
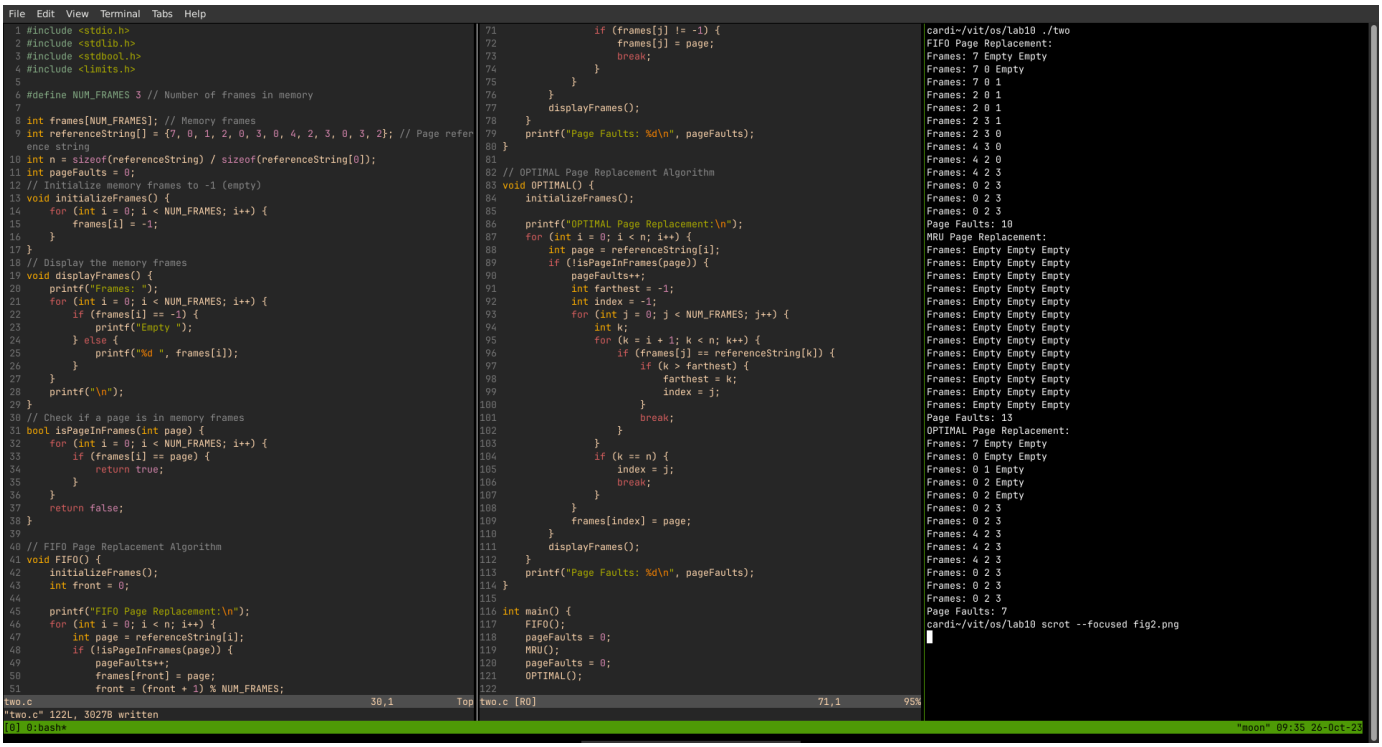
$22bps1059$

one

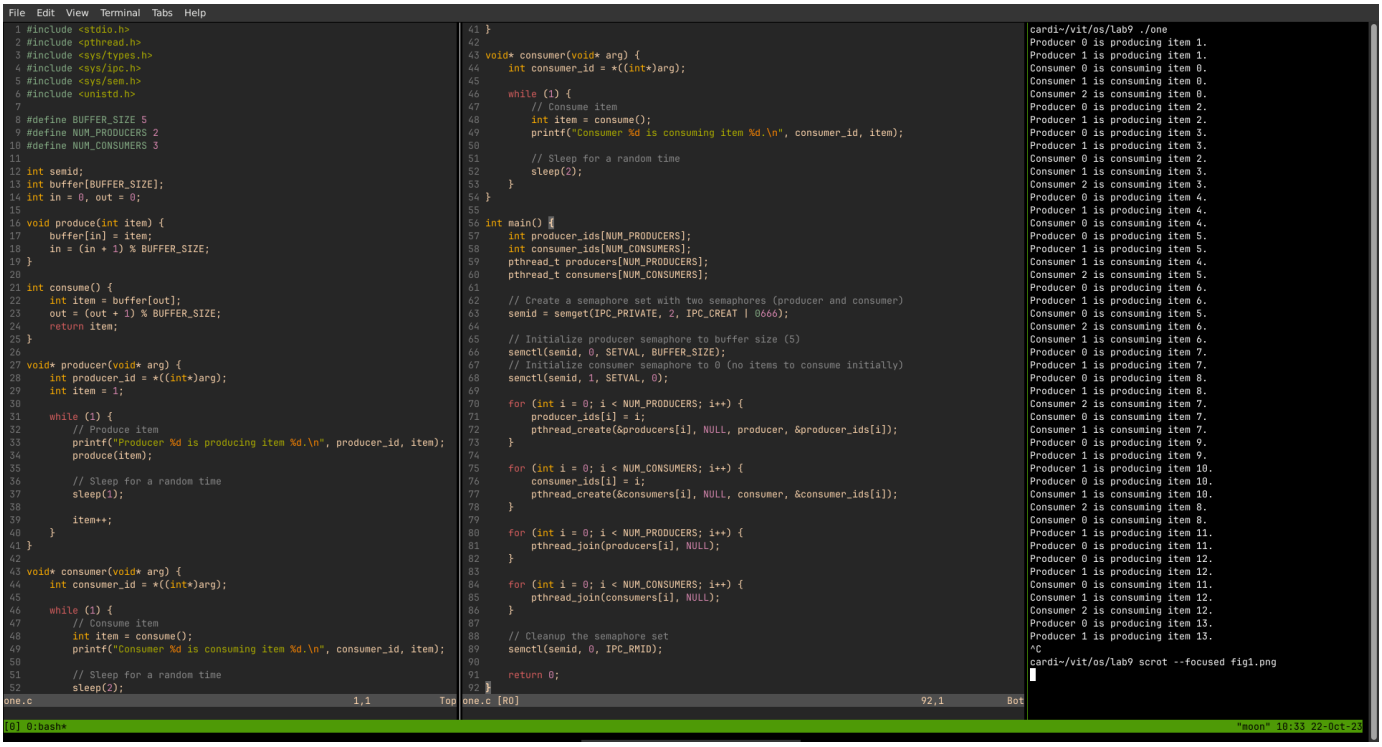


two

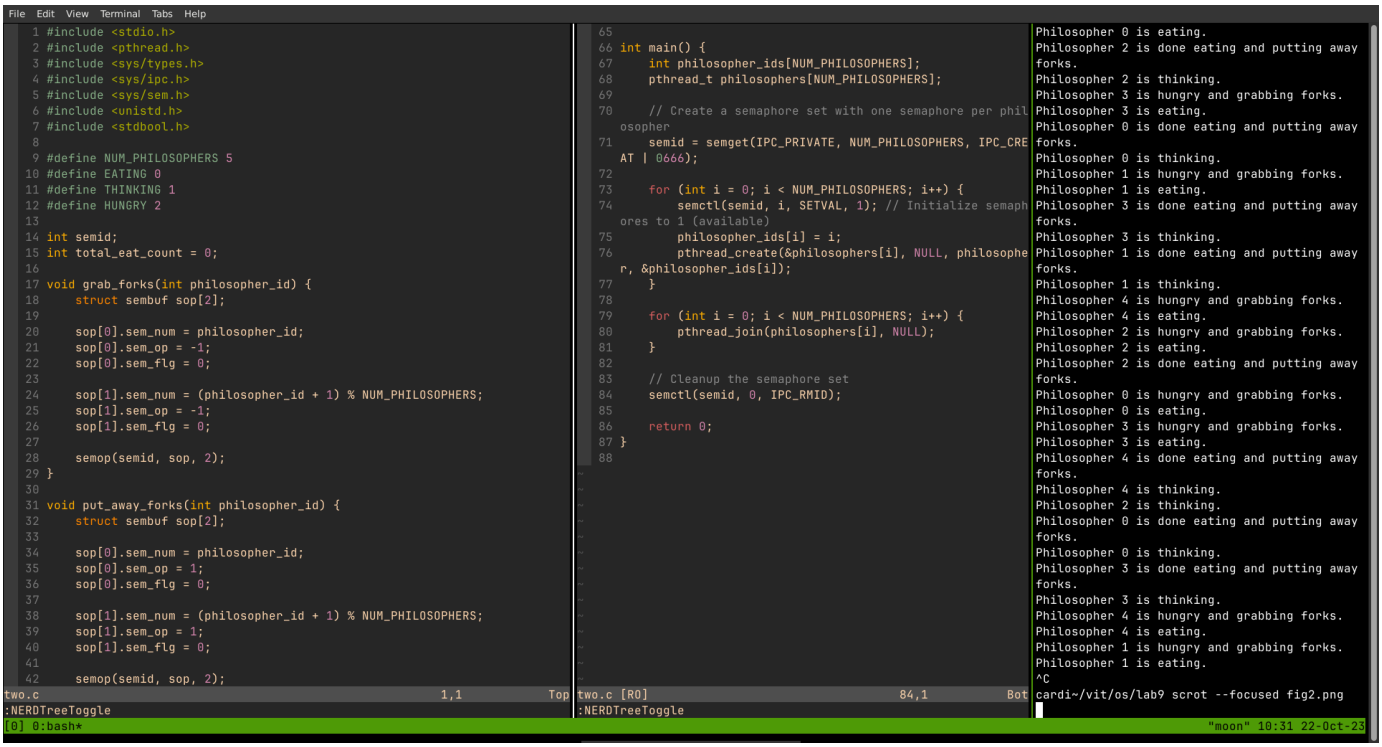

File Edit View Terminal Tabs Help
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include <limits.h>
#define NUM_FRAMES 3 // Number of frames in memory
int frames[NUM_FRAMES]; // Memory frames
int referenceString[] = {7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2}; // Page reference string
int n = sizeof(referenceString) / sizeof(referenceString[0]);
int pageFaults = 0;
// Initialize memory frames to -1 (empty)
void initializeFrames() {
for (int i = 0; i < NUM_FRAMES; i++) {
frames[i] = -1;
}
}
// Display the memory frames
void displayFrames() {
printf("Frames: ");
for (int i = 0; i < NUM_FRAMES; i++) {
if (frames[i] == -1) {
printf("Empty ");
} else {
printf("%d ", frames[i]);
}
}
printf("\n");
}
// Check if a page is in memory frames
bool isPageInFrames(int page) {
for (int i = 0; i < NUM_FRAMES; i++) {
if (frames[i] == page) {
return true;
}
}
return false;
}
// FIFO Page Replacement Algorithm
void FIFO() {
initializeFrames();
int front = 0;
printf("FIFO Page Replacement:\n");
for (int i = 0; i < n; i++) {
int page = referenceString[i];
if (!isPageInFrames(page)) {
pageFaults++;
frames[front] = page;
front = (front + 1) % NUM_FRAMES;
two.c 30,1 Top
"two.c" 122L, 3027B written
if (frames[j] != -1) {
frames[j] = page;
break;
}
}
displayFrames();
}
printf("Page Faults: %d\n", pageFaults);
}
// OPTIMAL Page Replacement Algorithm
void OPTIMAL() {
initializeFrames();
printf("OPTIMAL Page Replacement:\n");
for (int i = 0; i < n; i++) {
int page = referenceString[i];
if (!isPageInFrames(page)) {
pageFaults++;
int farthest = -1;
int index = -1;
for (int j = 0; j < NUM_FRAMES; j++) {
int k;
for (k = i + 1; k < n; k++) {
if (frames[j] == referenceString[k]) {
if (k > farthest) {
farthest = k;
index = j;
}
break;
}
}
if (k == n) {
index = j;
break;
}
}
frames[index] = page;
}
displayFrames();
}
printf("Page Faults: %d\n", pageFaults);
}
int main() {
FIFO();
pageFaults = 0;
MRU();
pageFaults = 0;
OPTIMAL();
two.c [RO] 71,1 95%
cardi~/vit/os/lab10 ./two
FIFO Page Replacement:
Frames: 7 Empty Empty
Frames: 7 0 Empty
Frames: 7 0 1
Frames: 2 0 1
Frames: 2 0 1
Frames: 2 3 1
Frames: 2 3 0
Frames: 4 3 0
Frames: 4 2 0
Frames: 4 2 3
Frames: 0 2 3
Frames: 0 2 3
Frames: 0 2 3
Page Faults: 10
MRU Page Replacement:
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Frames: Empty Empty Empty
Page Faults: 13
OPTIMAL Page Replacement:
Frames: 7 Empty Empty
Frames: 0 Empty Empty
Frames: 0 1 Empty
Frames: 0 2 Empty
Frames: 0 2 Empty
Frames: 0 2 3
Frames: 0 2 3
Frames: 4 2 3
Frames: 4 2 3
Frames: 4 2 3
Frames: 0 2 3
Frames: 0 2 3
Frames: 0 2 3
Page Faults: 7
cardi~/vit/os/lab10 scrot --focused fig2.png
[0] 0:bash* "moon" 09:35 26-Oct-23

$22bps1059$

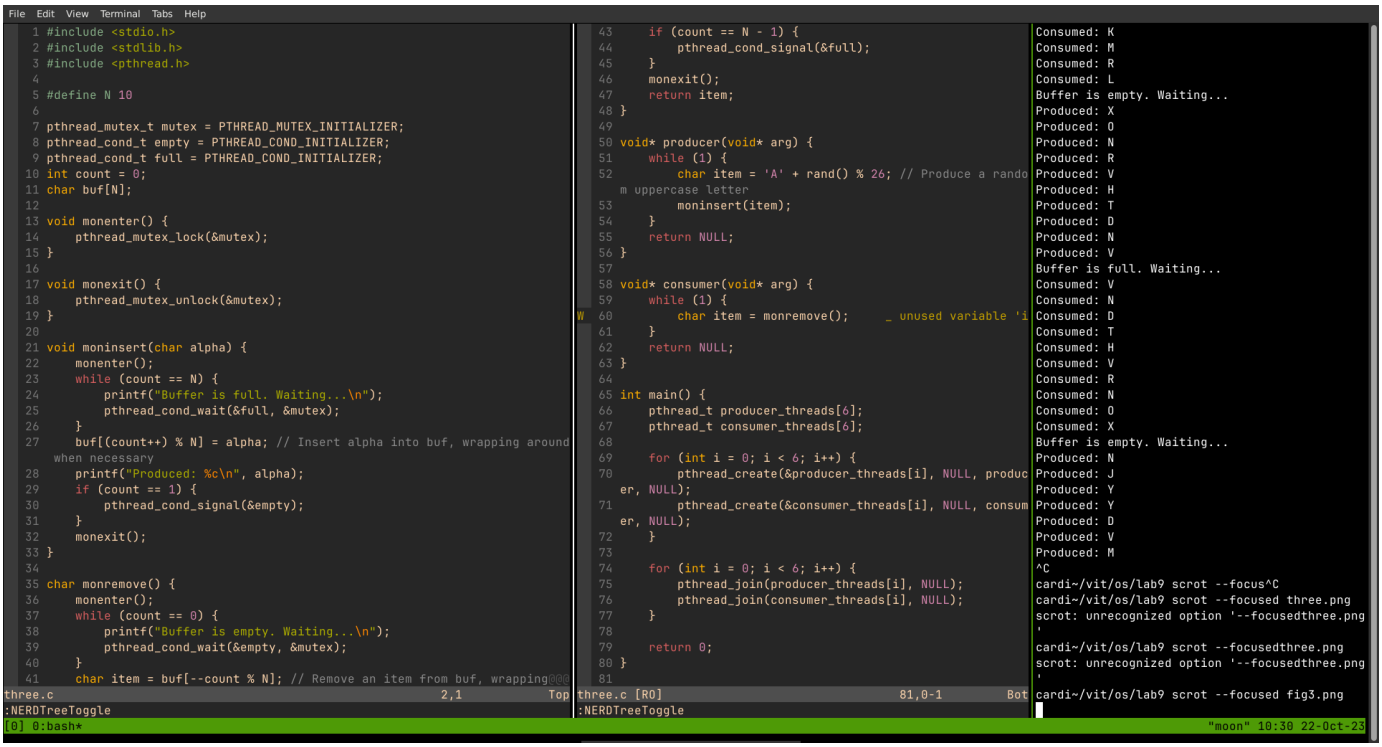
$first$



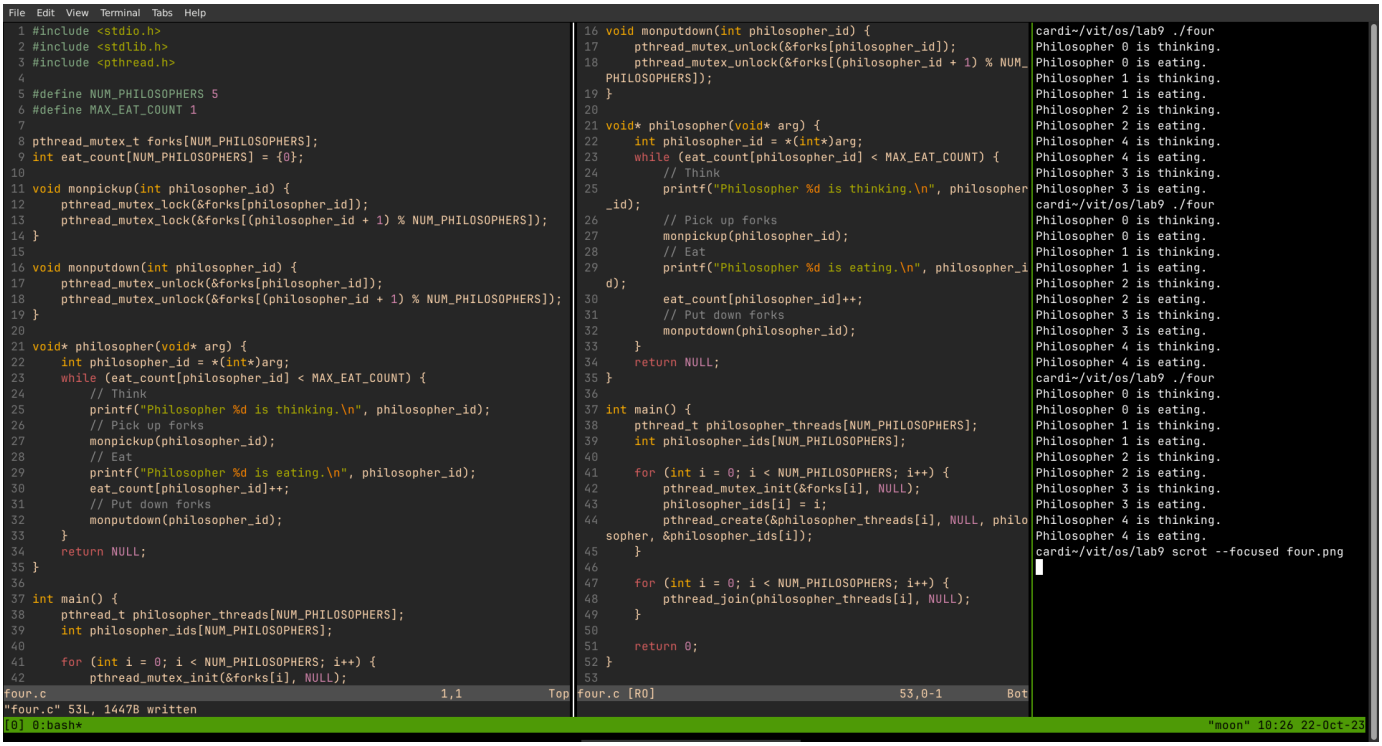
second


File Edit View Terminal Tabs Help
two.c 1,1 Top
:NERDTreeToggle
two.c [RO] 84,1 Bot
:NERDTreeToggle
[0] 0:bash*
"moon" 10:31 22-Oct-23
cardi~/vit/os/lab9 scrot --focused fig2.png
^C

third


File Edit View Terminal Tabs Help
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#define N 10
pthread_mutex_t mutex = PTHREAD_MUTEX_INITIALIZER;
pthread_cond_t empty = PTHREAD_COND_INITIALIZER;
pthread_cond_t full = PTHREAD_COND_INITIALIZER;
int count = 0;
char buf[N];
void monenter() {
pthread_mutex_lock(&mutex);
}
void monexit() {
pthread_mutex_unlock(&mutex);
}
void moninsert(char alpha) {
monenter();
while (count == N) {
printf("Buffer is full. Waiting...\n");
pthread_cond_wait(&full, &mutex);
}
buf[(count++) % N] = alpha; // Insert alpha into buf, wrapping around when necessary
printf("Produced: %c\n", alpha);
if (count == 1) {
pthread_cond_signal(&empty);
}
monexit();
}
char monremove() {
monenter();
while (count == 0) {
printf("Buffer is empty. Waiting...\n");
pthread_cond_wait(&empty, &mutex);
}
char item = buf[--count % N]; // Remove an item from buf, wrapping@@
three.c 2,1 Top
:NERDTreeToggle
if (count == N - 1) {
pthread_cond_signal(&full);
}
monexit();
return item;
}
void* producer(void* arg) {
while (1) {
char item = 'A' + rand() % 26; // Produce a random uppercase letter
moninsert(item);
}
return NULL;
}
void* consumer(void* arg) {
while (1) {
char item = monremove(); _ unused variable 'i
}
return NULL;
}
int main() {
pthread_t producer_threads[6];
pthread_t consumer_threads[6];
for (int i = 0; i < 6; i++) {
pthread_create(&producer_threads[i], NULL, producer, NULL);
pthread_create(&consumer_threads[i], NULL, consumer, NULL);
}
for (int i = 0; i < 6; i++) {
pthread_join(producer_threads[i], NULL);
pthread_join(consumer_threads[i], NULL);
}
return 0;
}
three.c [RO] 81,0-1 Bot
:NERDTreeToggle
Consumed: K
Consumed: M
Consumed: R
Consumed: L
Buffer is empty. Waiting...
Produced: X
Produced: O
Produced: N
Produced: R
Produced: V
Produced: H
Produced: T
Produced: D
Produced: N
Produced: V
Buffer is full. Waiting...
Consumed: V
Consumed: N
Consumed: D
Consumed: T
Consumed: H
Consumed: V
Consumed: R
Consumed: N
Consumed: O
Consumed: X
Buffer is empty. Waiting...
Produced: N
Produced: J
Produced: Y
Produced: Y
Produced: D
Produced: V
Produced: M
^C
cardi~/vit/os/lab9 scrot --focus^C
cardi~/vit/os/lab9 scrot --focused three.png
scrot: unrecognized option '--focusedthree.png'
cardi~/vit/os/lab9 scrot --focusedthree.png
scrot: unrecognized option '--focusedthree.png'
cardi~/vit/os/lab9 scrot --focused fig3.png
[0] 0:bash*
"moon" 10:30 22-Oct-23

fourth



```
File  Edit  View  Terminal  Tabs  Help
```

```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>

#define NUM_PHILOSOPHERS 5
#define MAX_EAT_COUNT 1

pthread_mutex_t forks[NUM_PHILOSOPHERS];
int eat_count[NUM_PHILOSOPHERS] = {0};

void monpickup(int philosopher_id) {
    pthread_mutex_lock(&forks[philosopher_id]);
    pthread_mutex_lock(&forks[(philosopher_id + 1) % NUM_PHILOSOPHERS]);
}

void monputdown(int philosopher_id) {
    pthread_mutex_unlock(&forks[philosopher_id]);
    pthread_mutex_unlock(&forks[(philosopher_id + 1) % NUM_PHILOSOPHERS]);
}

void* philosopher(void* arg) {
    int philosopher_id = *(int*)arg;
    while (eat_count[philosopher_id] < MAX_EAT_COUNT) {
        // Think
        printf("Philosopher %d is thinking.\n", philosopher_id);
        // Pick up forks
        monpickup(philosopher_id);
        // Eat
        printf("Philosopher %d is eating.\n", philosopher_id);
        eat_count[philosopher_id]++;
        // Put down forks
        monputdown(philosopher_id);
    }
    return NULL;
}

int main() {
    pthread_t philosopher_threads[NUM_PHILOSOPHERS];
    int philosopher_ids[NUM_PHILOSOPHERS];

    for (int i = 0; i < NUM_PHILOSOPHERS; i++) {
        pthread_mutex_init(&forks[i], NULL);
```

```
four.c                                                    1,1           Top
"four.c" 53L, 1447B written
```

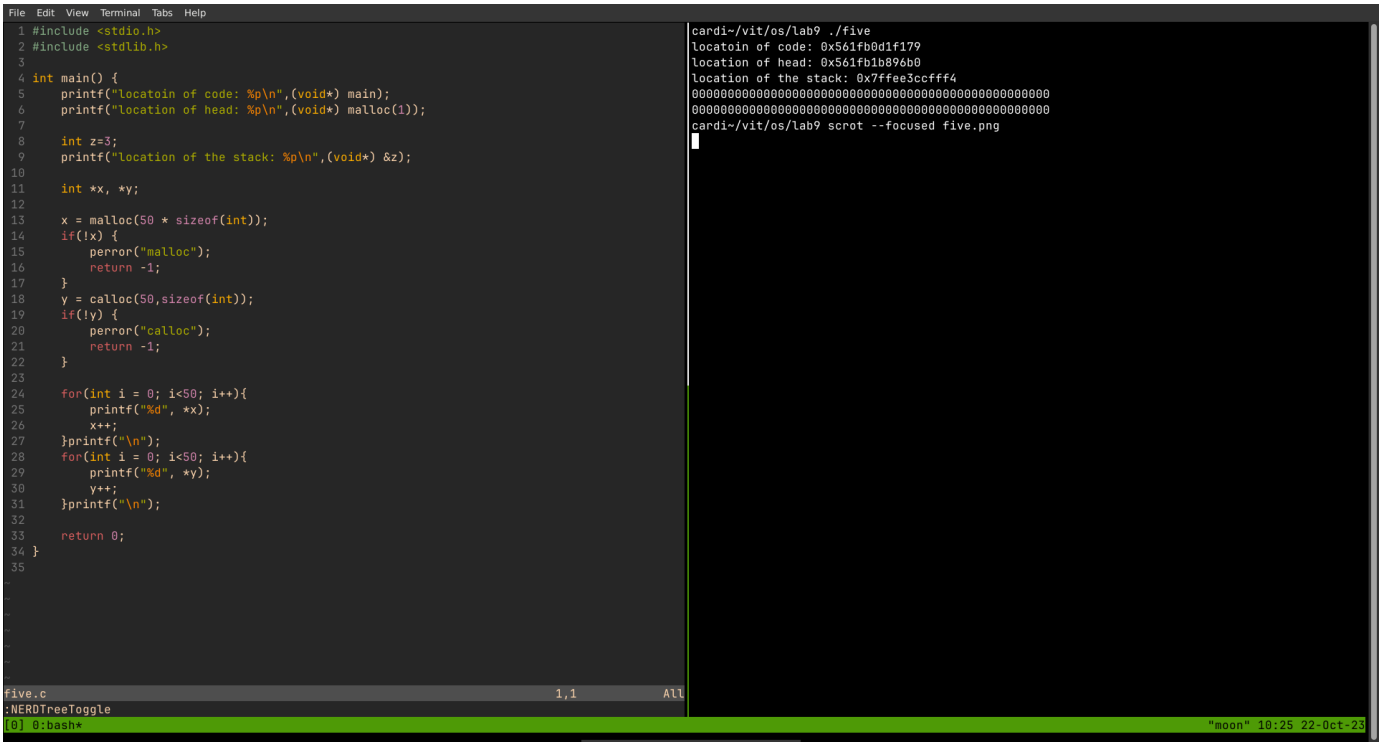
```c
void monputdown(int philosopher_id) {
    pthread_mutex_unlock(&forks[philosopher_id]);
    pthread_mutex_unlock(&forks[(philosopher_id + 1) % NUM_
PHILOSOPHERS]);
}

void* philosopher(void* arg) {
    int philosopher_id = *(int*)arg;
    while (eat_count[philosopher_id] < MAX_EAT_COUNT) {
        // Think
        printf("Philosopher %d is thinking.\n", philosopher
_id);
        // Pick up forks
        monpickup(philosopher_id);
        // Eat
        printf("Philosopher %d is eating.\n", philosopher_i
d);
        eat_count[philosopher_id]++;
        // Put down forks
        monputdown(philosopher_id);
    }
    return NULL;
}

int main() {
    pthread_t philosopher_threads[NUM_PHILOSOPHERS];
    int philosopher_ids[NUM_PHILOSOPHERS];

    for (int i = 0; i < NUM_PHILOSOPHERS; i++) {
        pthread_mutex_init(&forks[i], NULL);
        philosopher_ids[i] = i;
        pthread_create(&philosopher_threads[i], NULL, philo
sopher, &philosopher_ids[i]);
    }

    for (int i = 0; i < NUM_PHILOSOPHERS; i++) {
        pthread_join(philosopher_threads[i], NULL);
    }

    return 0;
}
```

```
four.c [RO]                                               53,0-1        Bot
```

```
cardi~/vit/os/lab9 ./four
Philosopher 0 is thinking.
Philosopher 0 is eating.
Philosopher 1 is thinking.
Philosopher 1 is eating.
Philosopher 2 is thinking.
Philosopher 2 is eating.
Philosopher 4 is thinking.
Philosopher 4 is eating.
Philosopher 3 is thinking.
Philosopher 3 is eating.
cardi~/vit/os/lab9 ./four
Philosopher 0 is thinking.
Philosopher 0 is eating.
Philosopher 1 is thinking.
Philosopher 1 is eating.
Philosopher 2 is thinking.
Philosopher 2 is eating.
Philosopher 3 is thinking.
Philosopher 3 is eating.
Philosopher 4 is thinking.
Philosopher 4 is eating.
cardi~/vit/os/lab9 ./four
Philosopher 0 is thinking.
Philosopher 0 is eating.
Philosopher 1 is thinking.
Philosopher 1 is eating.
Philosopher 2 is thinking.
Philosopher 2 is eating.
Philosopher 3 is thinking.
Philosopher 3 is eating.
Philosopher 4 is thinking.
Philosopher 4 is eating.
cardi~/vit/os/lab9 scrot --focused four.png
```

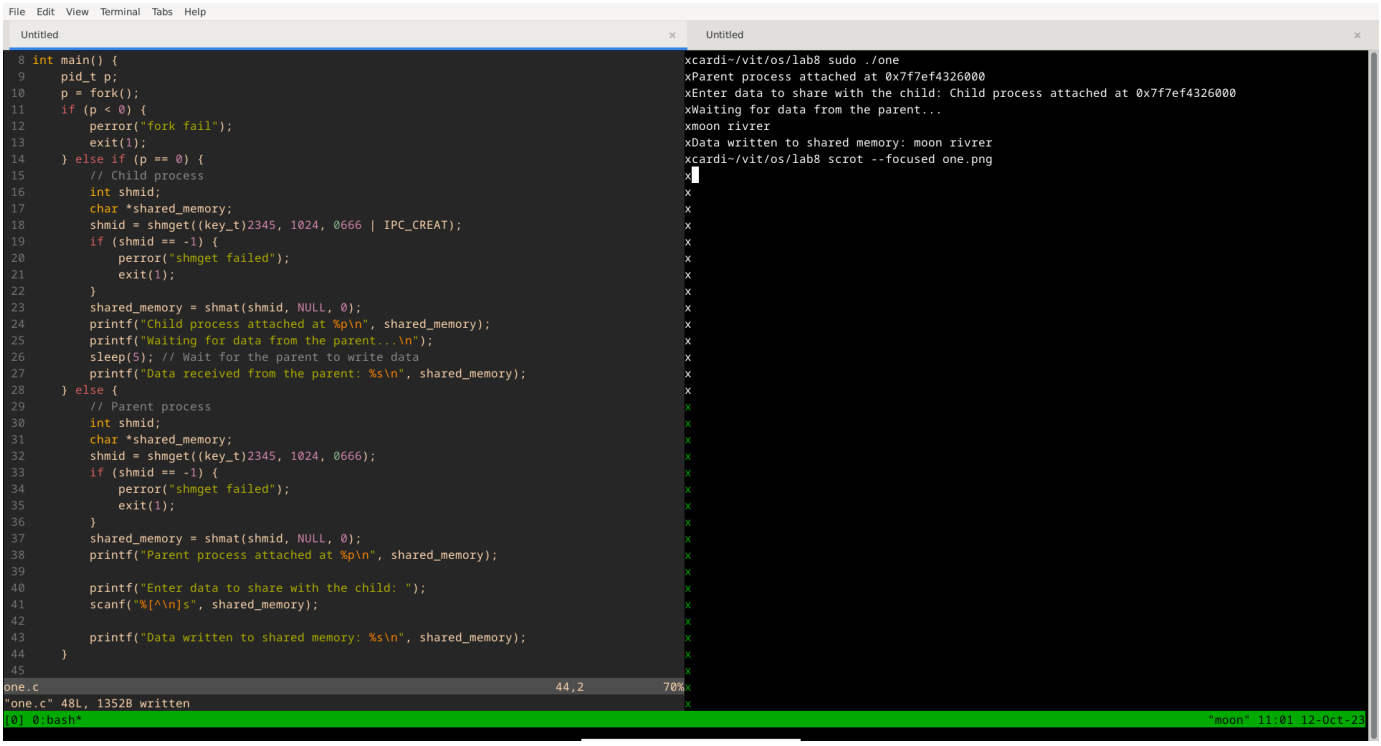
```
[0] 0:bash*                                                  "moon" 10:26 22-Oct-23
```

fifth

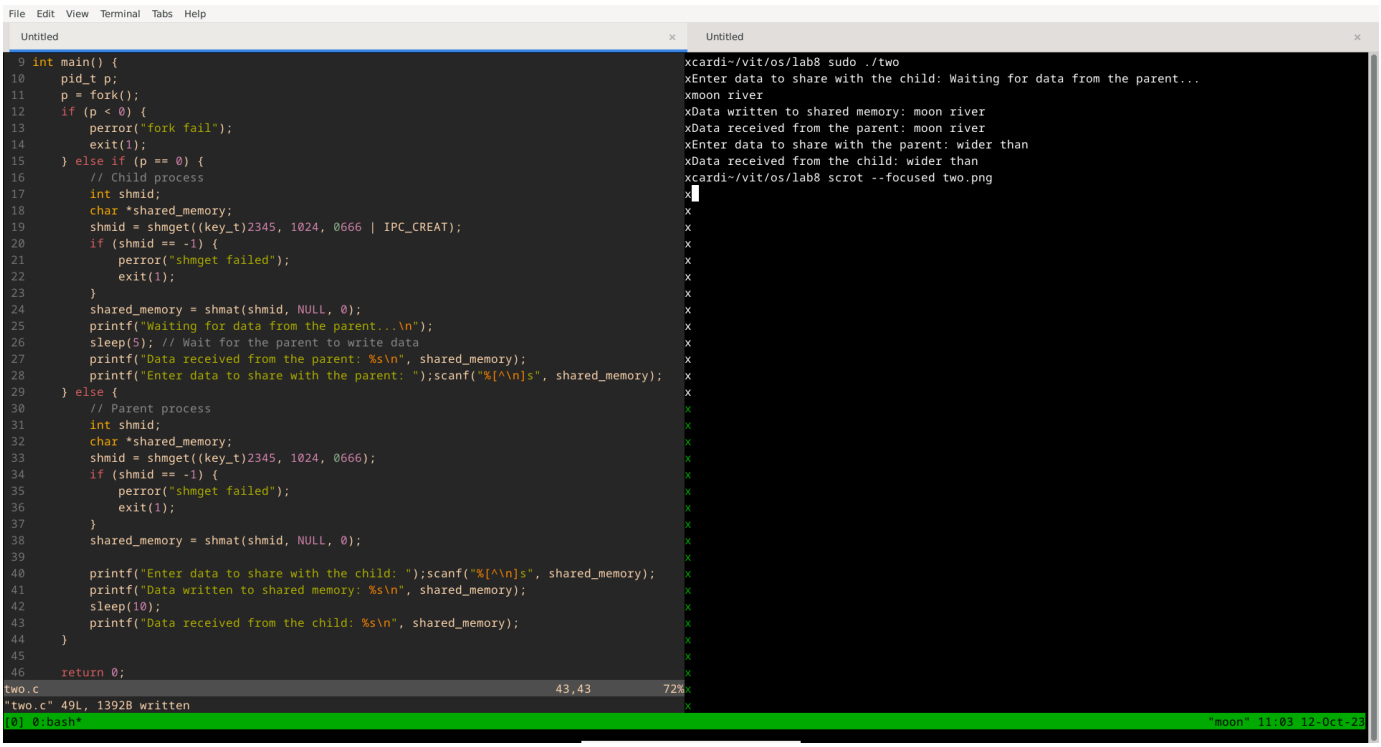

File Edit View Terminal Tabs Help
#include <stdio.h>
#include <stdlib.h>
int main() {
printf("locatoin of code: %p\n",(void*) main);
printf("location of head: %p\n",(void*) malloc(1));
int z=3;
printf("location of the stack: %p\n",(void*) &z);
int *x, *y;
x = malloc(50 * sizeof(int));
if(!x) {
perror("malloc");
return -1;
}
y = calloc(50,sizeof(int));
if(!y) {
perror("calloc");
return -1;
}
for(int i = 0; i<50; i++){
printf("%d", *x);
x++;
}printf("\n");
for(int i = 0; i<50; i++){
printf("%d", *y);
y++;
}printf("\n");
return 0;
}
five.c 1,1 All
:NERDTreeToggle
[0] 0:bash*
"moon" 10:25 22-Oct-23
cardi~/vit/os/lab9 ./five
locatoin of code: 0x561fb0d1f179
location of head: 0x561fb1b896b0
location of the stack: 0x7ffee3ccfff4
00000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000
cardi~/vit/os/lab9 scrot --focused five.png

$22bps1059$

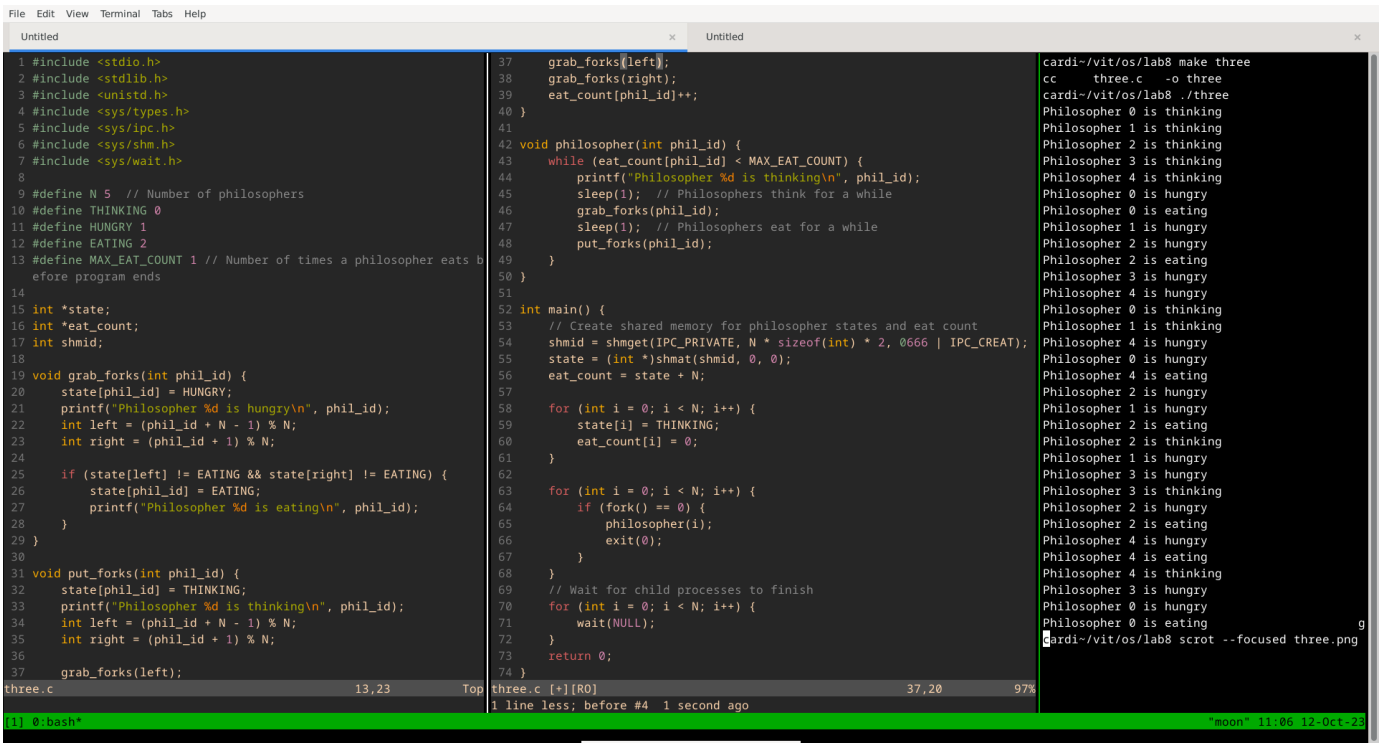
$first$



second


File Edit View Terminal Tabs Help
Untitled
Untitled
int main() {
pid_t p;
p = fork();
if (p < 0) {
perror("fork fail");
exit(1);
} else if (p == 0) {
// Child process
int shmid;
char *shared_memory;
shmid = shmget((key_t)2345, 1024, 0666 | IPC_CREAT);
if (shmid == -1) {
perror("shmget failed");
exit(1);
}
shared_memory = shmat(shmid, NULL, 0);
printf("Waiting for data from the parent...\n");
sleep(5); // Wait for the parent to write data
printf("Data received from the parent: %s\n", shared_memory);
printf("Enter data to share with the parent: ");scanf("%[^\n]s", shared_memory);
} else {
// Parent process
int shmid;
char *shared_memory;
shmid = shmget((key_t)2345, 1024, 0666);
if (shmid == -1) {
perror("shmget failed");
exit(1);
}
shared_memory = shmat(shmid, NULL, 0);
printf("Enter data to share with the child: ");scanf("%[^\n]s", shared_memory);
printf("Data written to shared memory: %s\n", shared_memory);
sleep(10);
printf("Data received from the child: %s\n", shared_memory);
}
return 0;
two.c 43,43 72%
"two.c" 49L, 1392B written
xcardi~/vit/os/lab8 sudo ./two
xEnter data to share with the child: Waiting for data from the parent...
xmoon river
xData written to shared memory: moon river
xData received from the parent: moon river
xEnter data to share with the parent: wider than
xData received from the child: wider than
xcardi~/vit/os/lab8 scrot --focused two.png
[0] 0:bash*
"moon" 11:03 12-Oct-23

third


File Edit View Terminal Tabs Help
Untitled
Untitled
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <sys/wait.h>
#define N 5 // Number of philosophers
#define THINKING 0
#define HUNGRY 1
#define EATING 2
#define MAX_EAT_COUNT 1 // Number of times a philosopher eats before program ends
int *state;
int *eat_count;
int shmid;
void grab_forks(int phil_id) {
state[phil_id] = HUNGRY;
printf("Philosopher %d is hungry\n", phil_id);
int left = (phil_id + N - 1) % N;
int right = (phil_id + 1) % N;
if (state[left] != EATING && state[right] != EATING) {
state[phil_id] = EATING;
printf("Philosopher %d is eating\n", phil_id);
}
}
void put_forks(int phil_id) {
state[phil_id] = THINKING;
printf("Philosopher %d is thinking\n", phil_id);
int left = (phil_id + N - 1) % N;
int right = (phil_id + 1) % N;
grab_forks(left);
three.c 13,23 Top
grab_forks(left);
grab_forks(right);
eat_count[phil_id]++;
}
void philosopher(int phil_id) {
while (eat_count[phil_id] < MAX_EAT_COUNT) {
printf("Philosopher %d is thinking\n", phil_id);
sleep(1); // Philosophers think for a while
grab_forks(phil_id);
sleep(1); // Philosophers eat for a while
put_forks(phil_id);
}
}
int main() {
// Create shared memory for philosopher states and eat count
shmid = shmget(IPC_PRIVATE, N * sizeof(int) * 2, 0666 | IPC_CREAT);
state = (int *)shmat(shmid, 0, 0);
eat_count = state + N;
for (int i = 0; i < N; i++) {
state[i] = THINKING;
eat_count[i] = 0;
}
for (int i = 0; i < N; i++) {
if (fork() == 0) {
philosopher(i);
exit(0);
}
}
// Wait for child processes to finish
for (int i = 0; i < N; i++) {
wait(NULL);
}
return 0;
}
three.c [+][RO] 37,20 97%
1 line less; before #4 1 second ago
cardi~/vit/os/lab8 make three
cc three.c -o three
cardi~/vit/os/lab8 ./three
Philosopher 0 is thinking
Philosopher 1 is thinking
Philosopher 2 is thinking
Philosopher 3 is thinking
Philosopher 4 is thinking
Philosopher 0 is hungry
Philosopher 0 is eating
Philosopher 1 is hungry
Philosopher 2 is hungry
Philosopher 2 is eating
Philosopher 3 is hungry
Philosopher 4 is hungry
Philosopher 0 is thinking
Philosopher 1 is thinking
Philosopher 4 is hungry
Philosopher 0 is hungry
Philosopher 4 is eating
Philosopher 2 is hungry
Philosopher 1 is hungry
Philosopher 2 is eating
Philosopher 2 is thinking
Philosopher 1 is hungry
Philosopher 3 is hungry
Philosopher 3 is thinking
Philosopher 2 is hungry
Philosopher 2 is eating
Philosopher 4 is hungry
Philosopher 4 is eating
Philosopher 4 is thinking
Philosopher 3 is hungry
Philosopher 0 is hungry
Philosopher 0 is eating
cardi~/vit/os/lab8 scrot --focused three.png
[1] 0:bash*
"moon" 11:06 12-Oct-23

$22bps1059$

first

File Edit View Search Terminal Help

```c
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <stdlib.h>
#include <sys/types.h>

int main() {
    int fd[2];
    pid_t childpid;
    char* string = "nobara";
    char readbuffer[20]; // Define a buffer for reading data
    ssize_t nbytes;     // Use ssize_t for read/write return values

    if (pipe(fd) == -1) {
        perror("Pipe");
        exit(1);
    }

    if ((childpid = fork()) == -1) {
        perror("Fork");
        exit(1);
    }

    if (childpid == 0) {
        close(fd[0]); // Close the read end of the pipe in the child
        write(fd[1], string, strlen(string) + 1);
        close(fd[1]); // Close the write end after writing
    } else {
        close(fd[1]); // Close the write end of the pipe in the parent
        nbytes = read(fd[0], readbuffer, sizeof(readbuffer));
        close(fd[0]); // Close the read end after reading

        if (nbytes < 0) {
            perror("Read");
            exit(1);
        } else if (nbytes == 0) {
            printf("No data received from child.\n");
        } else {
            printf("Received string: %s\n", readbuffer);
        }
    }

    return 0;
}
```

```
one.c                                                    1,1           Top
```

```
cardi~/coding/os/27seplab ./one
Received string: nobara
cardi~/coding/os/27seplab scrot --focused fig1.png
```

```
[5] 0:bash*                                  "thecuber-ThinkPad-E14" 14:36 01-Oct-23
```

second

File Edit View Search Terminal Help

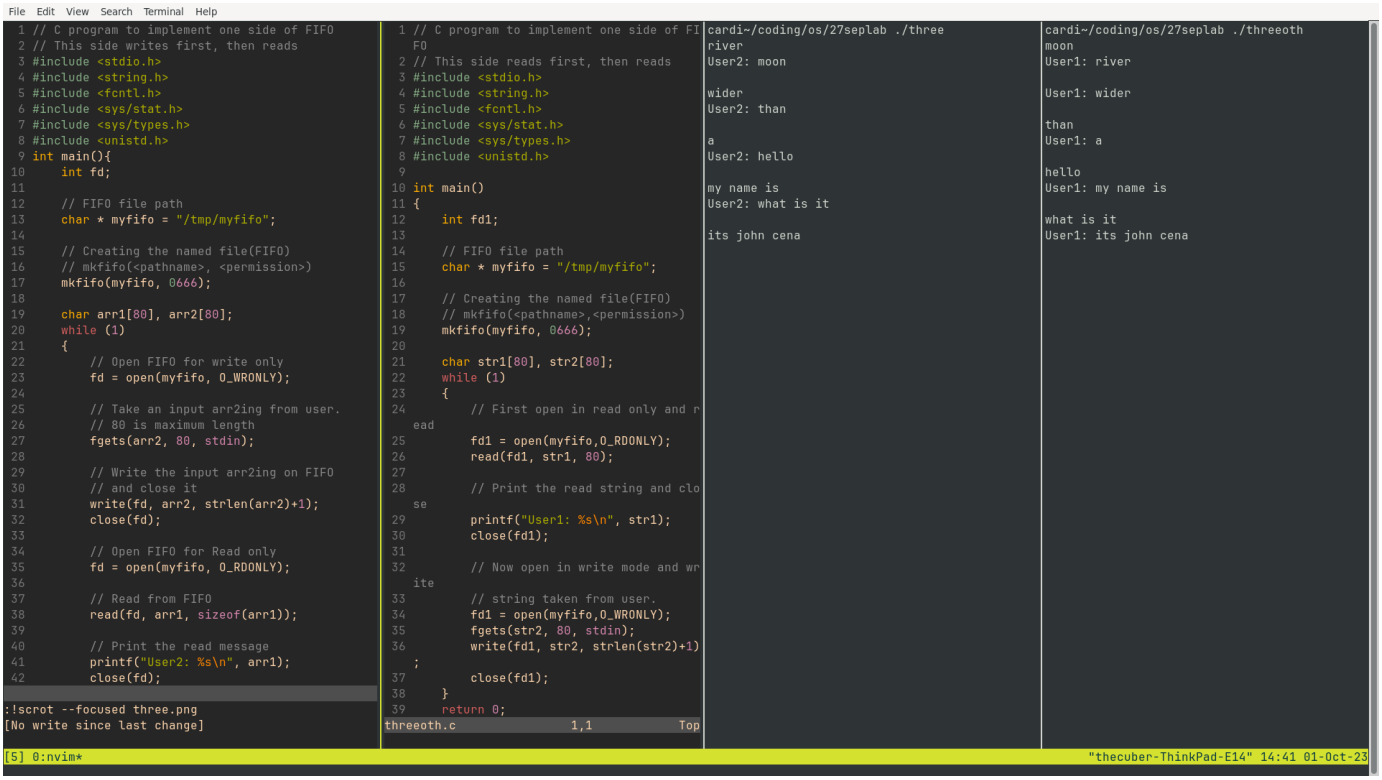
```c
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
//sem_t sync;
int main() {
    int ptc_pipe[2];
    int ctp_pipe[2];
    pid_t childpid;

    char* parentToChild = "loveptc";
    char* childToParent = "lovectp";
    char readbufferptc[20];
    char readbufferctp[20];
    ssize_t ptcbytes;
    ssize_t ctpbytes;

    if(pipe(ptc_pipe) == -1 || pipe(ctp_pipe) == -1) {
        perror("pipe");
        exit(1);
    }

    if((childpid = fork()) == -1) {
        perror("fork");
    }

    if(childpid == 0) {
        printf("child pid esta %d\n", getpid());

        //these will only be useful for parent
        close(ptc_pipe[1]); // close write end of ptc pipe
        close(ctp_pipe[0]); // close read end of ctp pipe

        ptcbytes = read(ptc_pipe[0], readbufferptc, sizeof(readbufferptc));

        if(ptcbytes < 0) {
            perror("Read");
            exit(1);
        } else if (ptcbytes == 0)  {
            printf("No data recieved from parent. \n");
        } else {
            printf("Recieved string in child: %s\n", readbufferptc);
        }
        close(ptc_pipe[0]); // closed read end of ptc as already read
        write(ctp_pipe[1], childToParent, strlen(childToParent) + 1);
        close(ctp_pipe[1]);// close write

    }
    else {
        printf("parent pid esta %d\n", getpid());

        //these will only be useful for child
        close(ctp_pipe[1]);
        close(ptc_pipe[0]);

        write(ptc_pipe[1], parentToChild, strlen(parentToChild) + 1);
        close(ptc_pipe[1]);// close write

        ctpbytes = read(ctp_pipe[0], readbufferctp, sizeof(readbufferctp));
        if(ctpbytes < 0) {
            perror("Read");
            exit(1);
        } else if (ctpbytes == 0)  {
            printf("No data recieved from child. \n");
        } else {
            printf("Recieved string in parent: %s\n", readbufferctp);
        }
        close(ctp_pipe[0]); // closed read end of ctp as already read

        wait(NULL);
    }
    return 0;
}
```

```
two.c                                                    1,1           Top
two.c [RO]                                               76,1          Bot
```

```
cardi~/coding/os/27seplab ./two
parent pid esta 553995
child pid esta 553996
Recieved string in child: loveptc
Recieved string in parent: lovectp
cardi~/coding/os/27seplab scrot --focused fig21.png
```

```
[5] 0:bash*                                  "thecuber-ThinkPad-E14" 14:38 01-Oct-23
```

third



```
File  Edit  View  Search  Terminal  Help
```

```c
// C program to implement one side of FIFO
// This side writes first, then reads
#include <stdio.h>
#include <string.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>
int main(){
    int fd;

    // FIFO file path
    char * myfifo = "/tmp/myfifo";

    // Creating the named file(FIFO)
    // mkfifo(<pathname>, <permission>)
    mkfifo(myfifo, 0666);

    char arr1[80], arr2[80];
    while (1)
    {
        // Open FIFO for write only
        fd = open(myfifo, O_WRONLY);

        // Take an input arr2ing from user.
        // 80 is maximum length
        fgets(arr2, 80, stdin);

        // Write the input arr2ing on FIFO
        // and close it
        write(fd, arr2, strlen(arr2)+1);
        close(fd);

        // Open FIFO for Read only
        fd = open(myfifo, O_RDONLY);

        // Read from FIFO
        read(fd, arr1, sizeof(arr1));

        // Print the read message
        printf("User2: %s\n", arr1);
        close(fd);
```

```
:!scrot --focused three.png
[No write since last change]
```

```c
// C program to implement one side of FIFO
// This side reads first, then reads
#include <stdio.h>
#include <string.h>
#include <fcntl.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>

int main()
{
    int fd1;

    // FIFO file path
    char * myfifo = "/tmp/myfifo";

    // Creating the named file(FIFO)
    // mkfifo(<pathname>,<permission>)
    mkfifo(myfifo, 0666);

    char str1[80], str2[80];
    while (1)
    {
        // First open in read only and read
        fd1 = open(myfifo,O_RDONLY);
        read(fd1, str1, 80);

        // Print the read string and close
        printf("User1: %s\n", str1);
        close(fd1);

        // Now open in write mode and write
        // string taken from user.
        fd1 = open(myfifo,O_WRONLY);
        fgets(str2, 80, stdin);
        write(fd1, str2, strlen(str2)+1);
        close(fd1);
    }
    return 0;
```

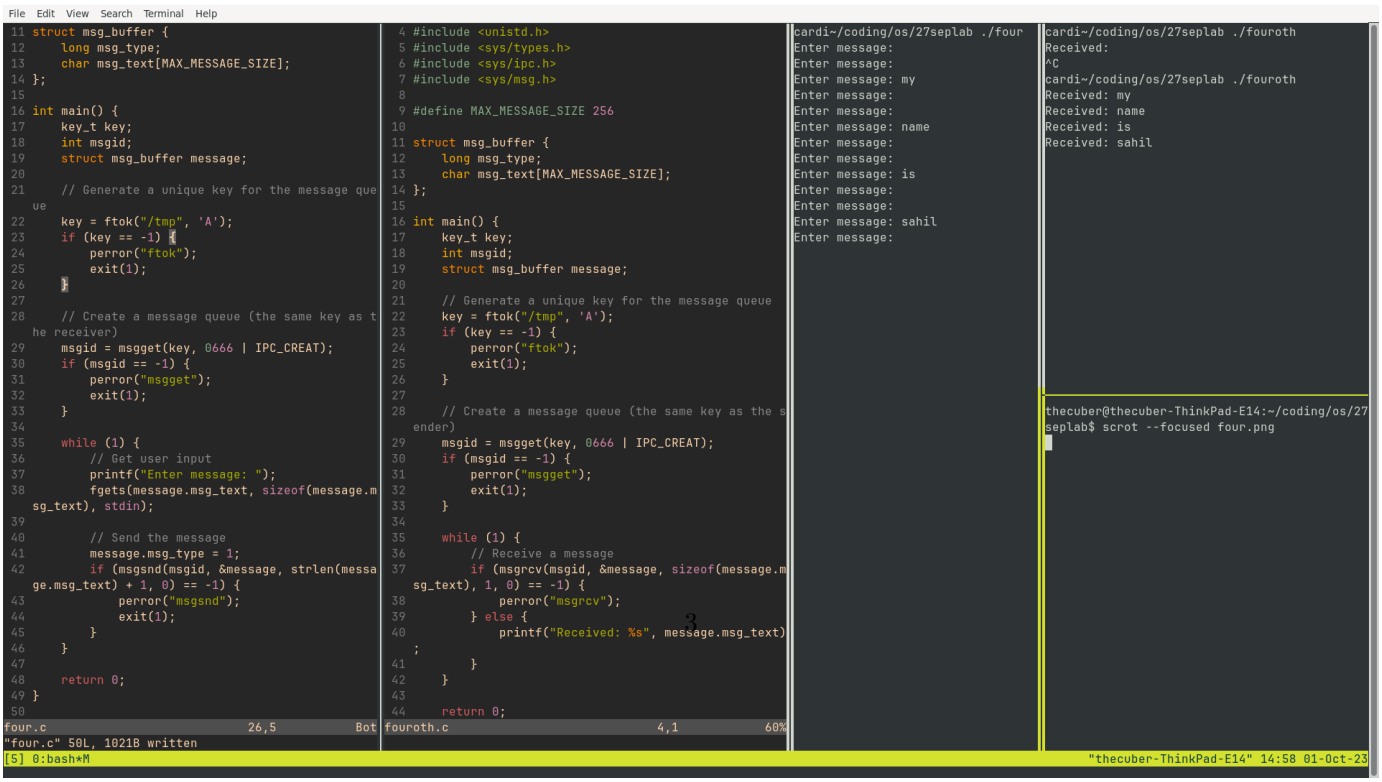
```
threeoth.c                    1,1           Top
```

```
cardi~/coding/os/27seplab ./three
river
User2: moon

wider
User2: than

a
User2: hello

my name is
User2: what is it

its john cena
```

```
cardi~/coding/os/27seplab ./threeoth
moon
User1: river

User1: wider

than
User1: a

hello
User1: my name is

what is it
User1: its john cena
```

```
[S] 0:nvim*                                  "thecuber-ThinkPad-E14" 14:41 01-Oct-23
```

fourth



```
File  Edit  View  Search  Terminal  Help
```

```c
struct msg_buffer {
    long msg_type;
    char msg_text[MAX_MESSAGE_SIZE];
};

int main() {
    key_t key;
    int msgid;
    struct msg_buffer message;

    // Generate a unique key for the message queue
    key = ftok("/tmp", 'A');
    if (key == -1) {
        perror("ftok");
        exit(1);
    }

    // Create a message queue (the same key as the receiver)
    msgid = msgget(key, 0666 | IPC_CREAT);
    if (msgid == -1) {
        perror("msgget");
        exit(1);
    }

    while (1) {
        // Get user input
        printf("Enter message: ");
        fgets(message.msg_text, sizeof(message.msg_text), stdin);

        // Send the message
        message.msg_type = 1;
        if (msgsnd(msgid, &message, strlen(message.msg_text) + 1, 0) == -1) {
            perror("msgsnd");
            exit(1);
        }
    }

    return 0;
}
```

```
four.c                        26,5          Bot
"four.c" 50L, 1021B written
```

```c
#include <unistd.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>

#define MAX_MESSAGE_SIZE 256

struct msg_buffer {
    long msg_type;
    char msg_text[MAX_MESSAGE_SIZE];
};

int main() {
    key_t key;
    int msgid;
    struct msg_buffer message;

    // Generate a unique key for the message queue
    key = ftok("/tmp", 'A');
    if (key == -1) {
        perror("ftok");
        exit(1);
    }

    // Create a message queue (the same key as the sender)
    msgid = msgget(key, 0666 | IPC_CREAT);
    if (msgid == -1) {
        perror("msgget");
        exit(1);
    }

    while (1) {
        // Receive a message
        if (msgrcv(msgid, &message, sizeof(message.msg_text), 1, 0) == -1) {
            perror("msgrcv");
        } else {
            printf("Received: %s", message.msg_text);
        }
    }

    return 0;
```

```
fouroth.c                     4,1           60%
```

```
cardi~/coding/os/27seplab ./four
Enter message:
Enter message:
Enter message: my
Enter message:
Enter message:
Enter message: name
Enter message:
Enter message:
Enter message: is
Enter message:
Enter message:
Enter message: sahil
Enter message:
```

```
cardi~/coding/os/27seplab ./fouroth
Received:
^C
cardi~/coding/os/27seplab ./fouroth
Received: my
Received: name
Received: is
Received: sahil
```

```
thecuber@thecuber-ThinkPad-E14:~/coding/os/27seplab$ scrot --focused four.png
```

```
[S] 0:bash*M                                 "thecuber-ThinkPad-E14" 14:58 01-Oct-23
```

$22bps1059$

first

File Edit View Search Terminal Help

```c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <pthread.h>

void userfunction() {
    fflush(stdout);
    printf("just a user function\n");
}
void* worker() {
    //calls a system call sleep
    userfunction();
    printf("This is thread1 id: %ld\n", pthread_self());
    sleep(1);
    pthread_exit(0);
}

void* worker2() {
    printf("This is thread2 id: %ld\n", pthread_self());
    sleep(2);
    pthread_exit(0);
}
int main() {
    pthread_t t1;

    int start = clock();
    if( pthread_create(&t1, NULL, worker, NULL ) != 0 ) {
        perror("cannot create thread\n");
        exit(1);
    }
    int end = clock();
    printf("time taken by thread 1 is %d\n", end - start);

    pthread_t t2;
    int start1 = clock();
    if( pthread_create(&t2, NULL, worker2, NULL) !=0) {
        perror("error creating thread 2");
        exit(1);
    }
    int end1 = clock();
    printf("time taken by thread 2 is %d\n", end1 - start1);

    pthread_join(t1,NULL);
    pthread_join(t2,NULL);    return 0;}
```

```
onetime.c                                                    1,1           All
```

```
cardi~/coding/os/lab20sep_lab6 ./onetime
time taken by thread 1 is 147
just a user function
This is thread1 id: 140447376733760
time taken by thread 2 is 213
This is thread2 id: 140447368341056
cardi~/coding/os/lab20sep_lab6 scrot --focused one.png
```

```
[0] 0:bash*                                          "thecuber-ThinkPad-E14" 15:46 29-Sep-23
```

second

File Edit View Search Terminal Help

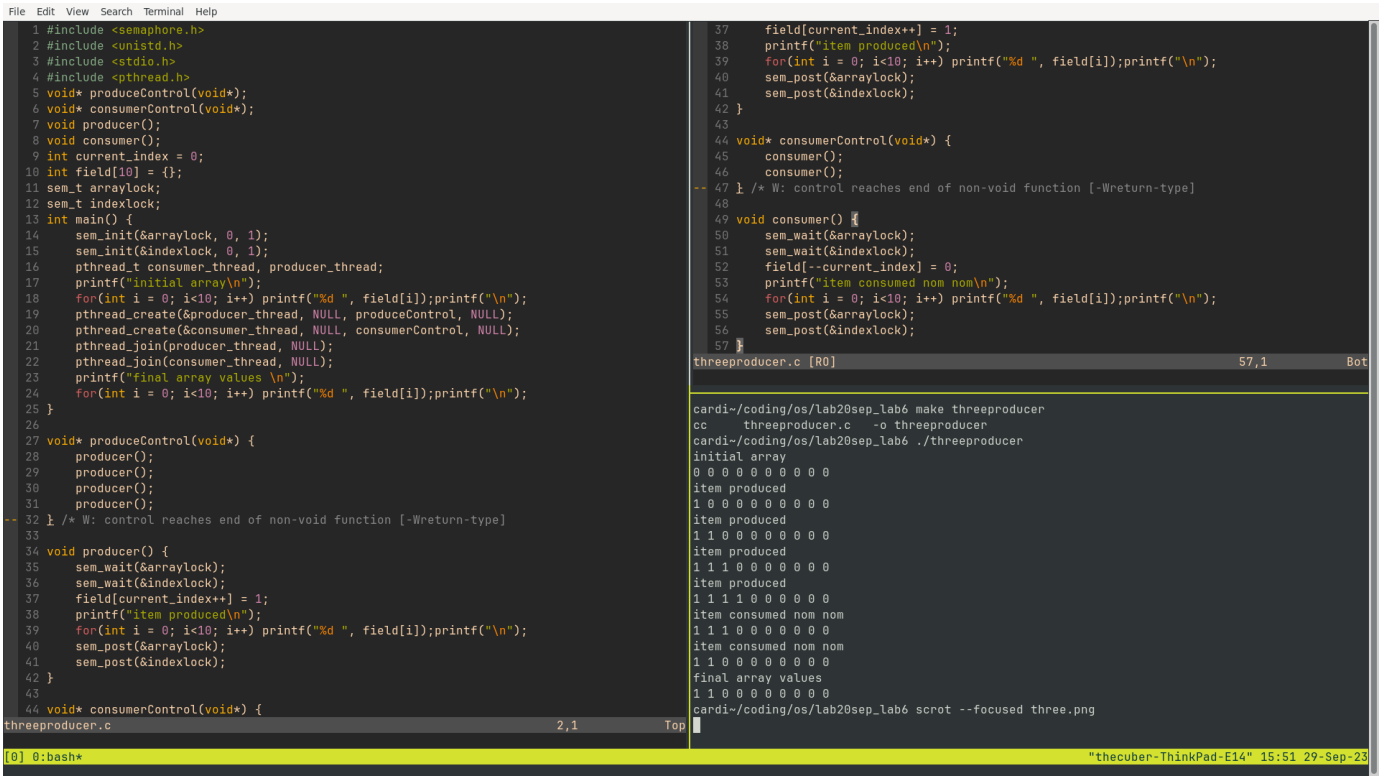
```c
#include <pthread.h>
#include <stdio.h>
#include <semaphore.h>
#include <unistd.h>

void* f1();
void* f2();

int shared = 1;
sem_t s;

int main() {
    sem_init(&s,0,1);
    pthread_t thread1,thread2;
    pthread_create(&thread1,NULL,f1,NULL);
    pthread_create(&thread2,NULL,f2,NULL);
    pthread_join(thread1,NULL);
    pthread_join(thread2,NULL);
    printf("Final value of shares is %d\n", shared);
}
void* f1() {
    int x;
    sem_wait(&s);
    x = shared;
    printf("Thread1 reads the value as %d\n", x);
    x++;
    printf("local updation by thread 1: %d\n", x);
    sleep(1);
    shared=x;
    printf("Value of shared variable updated by thread 1 is %d\n", shared);
    sem_post(&s);
}
void* f2() {
    int y;
    sem_wait(&s);
    y = shared;
    printf("THread 2 reads the value of %d\n", y);
    y--;
    printf("local updation by thread2: %d\n", y);
    sleep(1);
    shared = y;
    printf("value of shared variable updated by thread2 is %d\n", shared);
    sem_post(&s);
}
```

```
twosemaphore.c [+]                                           42,1          All
:ALEDisable
```

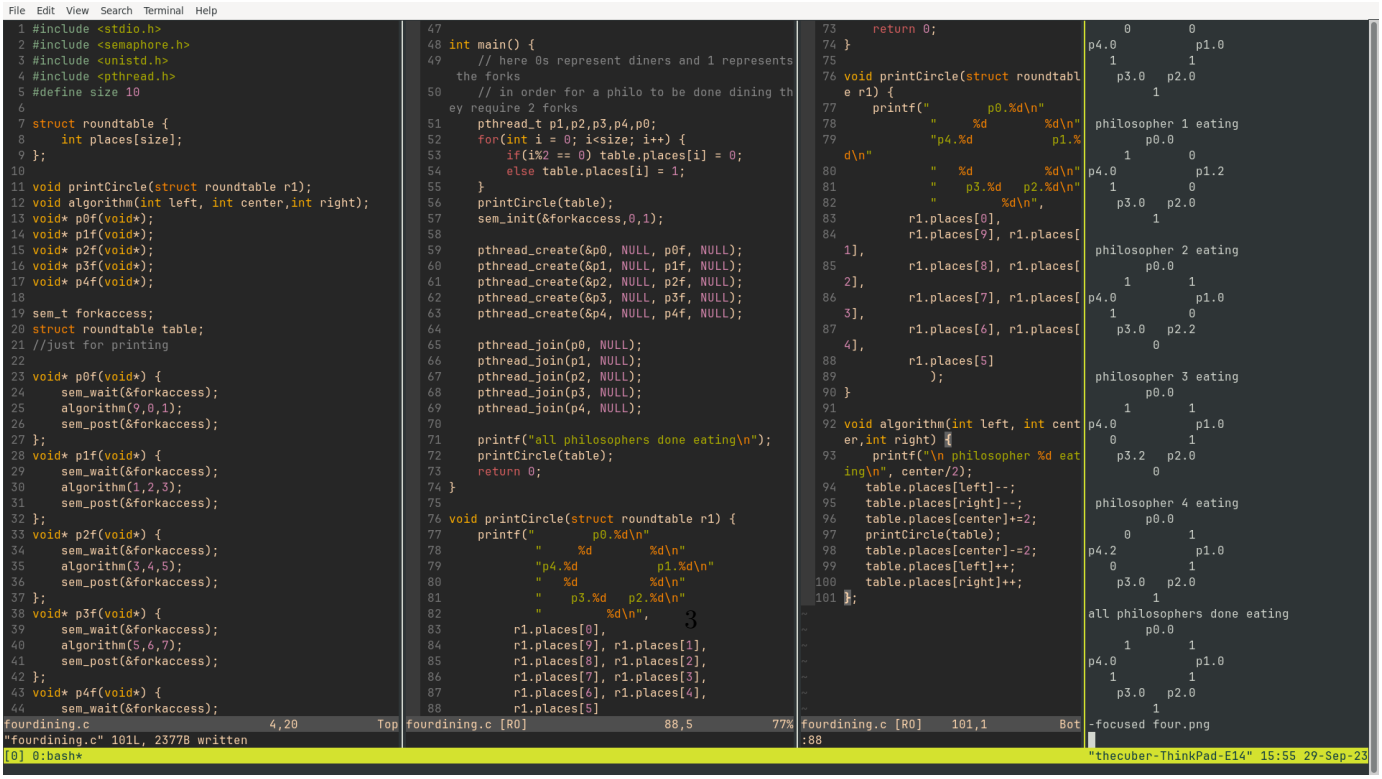
```
cardi~/coding/os/lab20sep_lab6 make twosemaphore
cc     twosemaphore.c   -o twosemaphore
cardi~/coding/os/lab20sep_lab6 ./twosemaphore
Thread1 reads the value as 1
local updation by thread 1: 2
Value of shared variable updated by thread 1 is 2
THread 2 reads the value of 2
local updation by thread2: 1
value of shared variable updated by thread2 is 1
Final value of shares is 1
cardi~/coding/os/lab20sep_lab6 scrot --focused two.png
```

```
[0] 0:bash*                                          "thecuber-ThinkPad-E14" 15:48 29-Sep-23
```

third



fourth



$22bps1059$

$first - sigint$

File Edit View Search Terminal Help

```c
#include <stdio.h>
#include <signal.h>
#include <sys/wait.h>
#include <unistd.h>
#include <stdlib.h>
#define SIGINT 2

void  siginthandler() {
    printf("handled\n");
    exit(0);
}

int main() {
    signal(SIGINT,siginthandler);
    while(1) {
        printf("Hello world\n");
        sleep(1);
    }
    return 0;
}
```

```
22BPS1059>./sigint2
Hello world
Hello world
Hello world
Hello world
Hello world
Hello world
Hello world
Hello world
^Chandled
22BPS1059>
```

sigint2.c 1,1 All

[0] 0:bash* "thecuber-ThinkPad-E14" 21:49 05-Sep-23

$second - sigint, sighup, sigquit$

File Edit View Search Terminal Help

```c
#include <signal.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/wait.h>
void sighup(); void sigint(); void sigquit();

void main() {
    int pid;
    pid = fork();
    if(pid == 0) {
        signal(SIGHUP,sighup);
        signal(SIGQUIT,sigint);
        signal(SIGINT,sigint);
        signal(SIGQUIT,sigquit);

        for(;;);
    }
    else {
        printf("\n parent: sending sighup\n\n");
        kill(pid,SIGHUP);
        sleep(3);
        printf("\n parent: sending sigint\n\n");
        kill(pid,SIGINT);
        sleep(3);
        printf("\n parent: sending sigquit\n\n");
        kill(pid,SIGQUIT);
        sleep(3);
    }
}

void sighup() {
    printf("CHILD: i have recieved a SIGHUP\n");
    fflush(stdout);
}

void sigint() {
    printf("CHILD: i have recieved a sigint\n");
    fflush(stdout);
}

void sigquit() {
    printf("CHILD: i have recieved a sigquit\n");
    fflush(stdout);
}
```

```
22BPS1059>./sigint

 parent: sending sighup

CHILD: i have recieved a SIGHUP

 parent: sending sigint

CHILD: i have recieved a sigint

 parent: sending sigquit

CHILD: i have recieved a sigquit
22BPS1059>
```

sigint.c 28,13 Bot

"sigint.c" 46L, 989B written

[0] 0:bash* "thecuber-ThinkPad-E14" 21:45 05-Sep-23

$third - roundrobin$

File Edit View Search Terminal Help

```c
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <signal.h>
#include <sys/types.h>
#include <sys/wait.h>
#define MAX_PROCESSES 5
#define TIME_QUANTUM 2
int completed = 0;
void execute(int pid, int time_slice) {
    printf("Executing Process %d for %d seconds\n", pid, time_slice);
    sleep(time_slice);
}
int main() {
    int n = 5; // Number of processes
    int arrival_time[] = {0, 1, 2, 3, 4}; /* W: unused variable 'arrival_time' [-Wunused-
    int burst_time[] = {6, 2, 8, 3, 5};
    int remaining_time[MAX_PROCESSES];
    int child_pids[MAX_PROCESSES];

    for (int i = 0; i < n; i++) {
        remaining_time[i] = burst_time[i];
    }

    int current_process = 0; /* W: unused variable 'current_process' [-Wunused-variable]
    for (int i = 0; i < n; i++) {
        pid_t child_pid = fork();

        if (child_pid == -1) {
            perror("fork");
            exit(1);
        }
        else if (child_pid == 0) {
            while (remaining_time[i] > 0) {
                int time_slice = (remaining_time[i] < TIME_QUANTUM) ? remaining_time[i] :
 TIME_QUANTUM;
                execute(i + 1, time_slice);
                remaining_time[i] -= time_slice;

                if (remaining_time[i] > 0) {
                    printf("Process %d preempted. Remaining time: %d seconds\n", i + 1, r
emaining_time[i]);
                }
            }
            exit(0);
        }
        else {
            child_pids[i] = child_pid;
        }
    }
    for (int i = 0; i < n; i++) {
        waitpid(child_pids[i], NULL, 0);
        printf("Parent Process: Child Process %d completed.\n", child_pids[i]);
    }

}
```

roundrobin.c 14,9 Top

"roundrobin.c" 57L, 1552B written

```
22BPS1059>./roundrobin
Executing Process 1 for 2 seconds
Executing Process 2 for 2 seconds
Executing Process 3 for 2 seconds
Executing Process 4 for 2 seconds
Executing Process 5 for 2 seconds
Process 4 preempted. Remaining time: 1 seconds
Process 1 preempted. Remaining time: 4 seconds
Process 3 preempted. Remaining time: 6 seconds
Process 5 preempted. Remaining time: 3 seconds
Executing Process 4 for 1 seconds
Executing Process 1 for 2 seconds
Executing Process 3 for 2 seconds
Executing Process 5 for 2 seconds
Process 3 preempted. Remaining time: 4 seconds
Process 1 preempted. Remaining time: 2 seconds
Process 5 preempted. Remaining time: 1 seconds
Executing Process 3 for 2 seconds
Executing Process 5 for 1 seconds
Executing Process 1 for 2 seconds
Process 3 preempted. Remaining time: 2 seconds
Executing Process 3 for 2 seconds
Parent Process: Child Process 336688 completed.
Parent Process: Child Process 336689 completed.
Parent Process: Child Process 336690 completed.
Parent Process: Child Process 336691 completed.
Parent Process: Child Process 336692 completed.
22BPS1059>
```

roundrobin.c [RO] 45,0-1 93%

[0] 0:bash* "thecuber-ThinkPad-E14" 21:37 05-Sep-23

$fourth - srjf$

File Edit View Search Terminal Help

```c
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <signal.h>
#include <sys/types.h>
#include <sys/wait.h>
#define MAX_PROCESSES 5
int completed = 0;

void execute(int pid, int time_slice) {
    printf("Executing Process %d for %d seconds\n", pid, time_slice);
    sleep(time_slice);
}
int main() {
    int n = 5;
    int arrival_time[] = {0, 1, 2, 3, 4};
    int burst_time[] = {6, 2, 8, 3, 5};
    int remaining_time[MAX_PROCESSES];
    int child_pids[MAX_PROCESSES];
    for (int i = 0; i < n; i++) remaining_time[i] = burst_time[i];
    int current_time = 0;

    for (int i = 0; i < n; i++) {
        pid_t child_pid = fork();

        if (child_pid == -1) {
            perror("fork");
            exit(1);
        } else if (child_pid == 0) { // Child process
            while (remaining_time[i] > 0 && !completed) {
                int time_slice = 1;
                if (remaining_time[i] < 1) {
                    time_slice = remaining_time[i];
                }

                execute(i + 1, time_slice);
                remaining_time[i] -= time_slice;
            }

            exit(0);
        } else { // Parent process
            child_pids[i] = child_pid;
        }
    }
    sleep(2);
    for (int i = 0; i < n; i++) {
        kill(child_pids[i], SIGHUP);
    }
    for (int i = 0; i < n; i++) {
        int status;
        waitpid(child_pids[i], &status, 0);
        if (WIFSIGNALED(status) && WTERMSIG(status) == SIGHUP) {
            printf("Parent Process: Child Process %d terminated.\n", child_pids[i]);
        }
    }
    return 0;
}
```

srjf.c 21,25 Top

"srjf.c" 58L, 1587B written

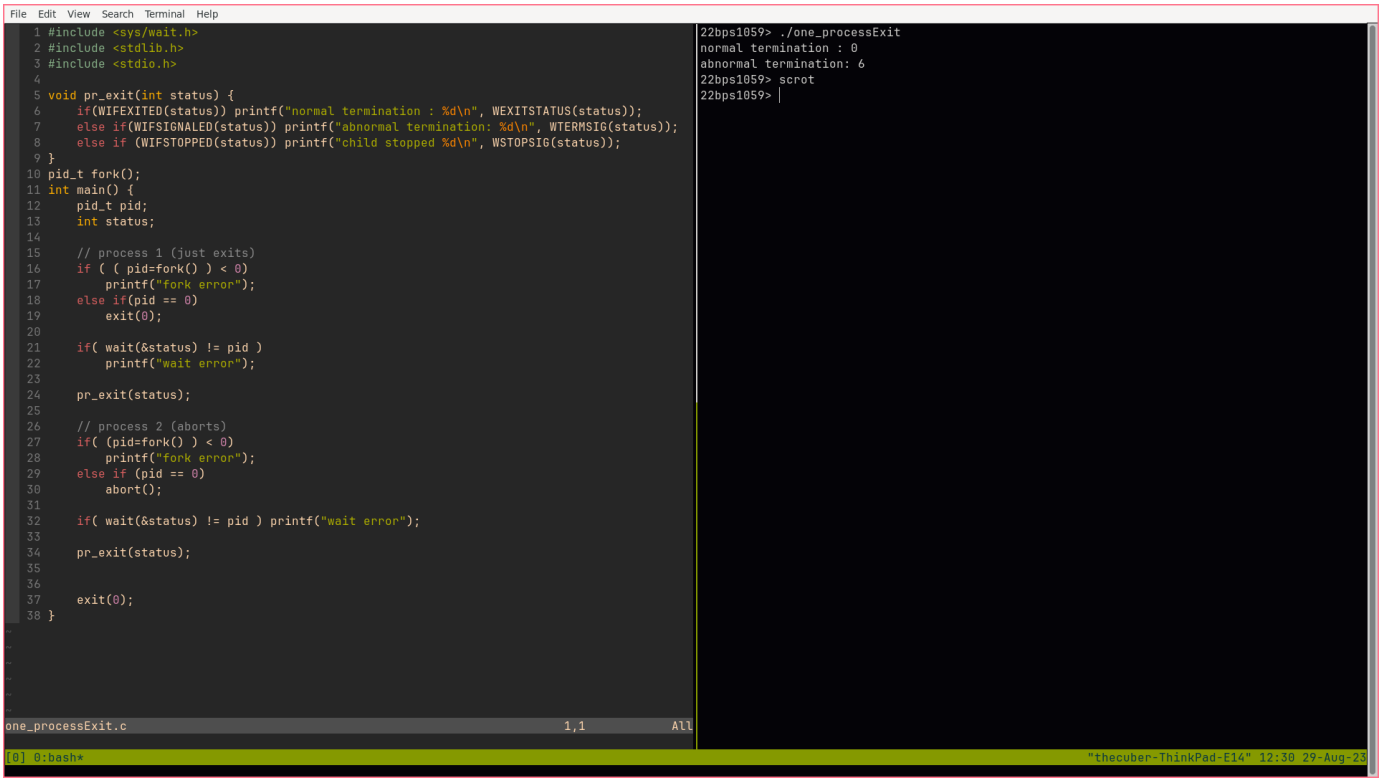
srjf.c [RO] 57,1 97%

```
22BPS1059>./srjf
Executing Process 1 for 1 seconds
Executing Process 2 for 1 seconds
Executing Process 3 for 1 seconds
Executing Process 4 for 1 seconds
Executing Process 5 for 1 seconds
Executing Process 1 for 1 seconds
Executing Process 2 for 1 seconds
Executing Process 3 for 1 seconds
Executing Process 4 for 1 seconds
Executing Process 5 for 1 seconds
Parent Process: Child Process 336938 terminated.
Parent Process: Child Process 336939 terminated.
Parent Process: Child Process 336940 terminated.
Parent Process: Child Process 336941 terminated.
Parent Process: Child Process 336942 terminated.
22BPS1059>
```

[0] 0:bash* "thecuber-ThinkPad-E14" 21:39 05-Sep-23

$22bps1059$

first



```
File  Edit  View  Search  Terminal  Help
```

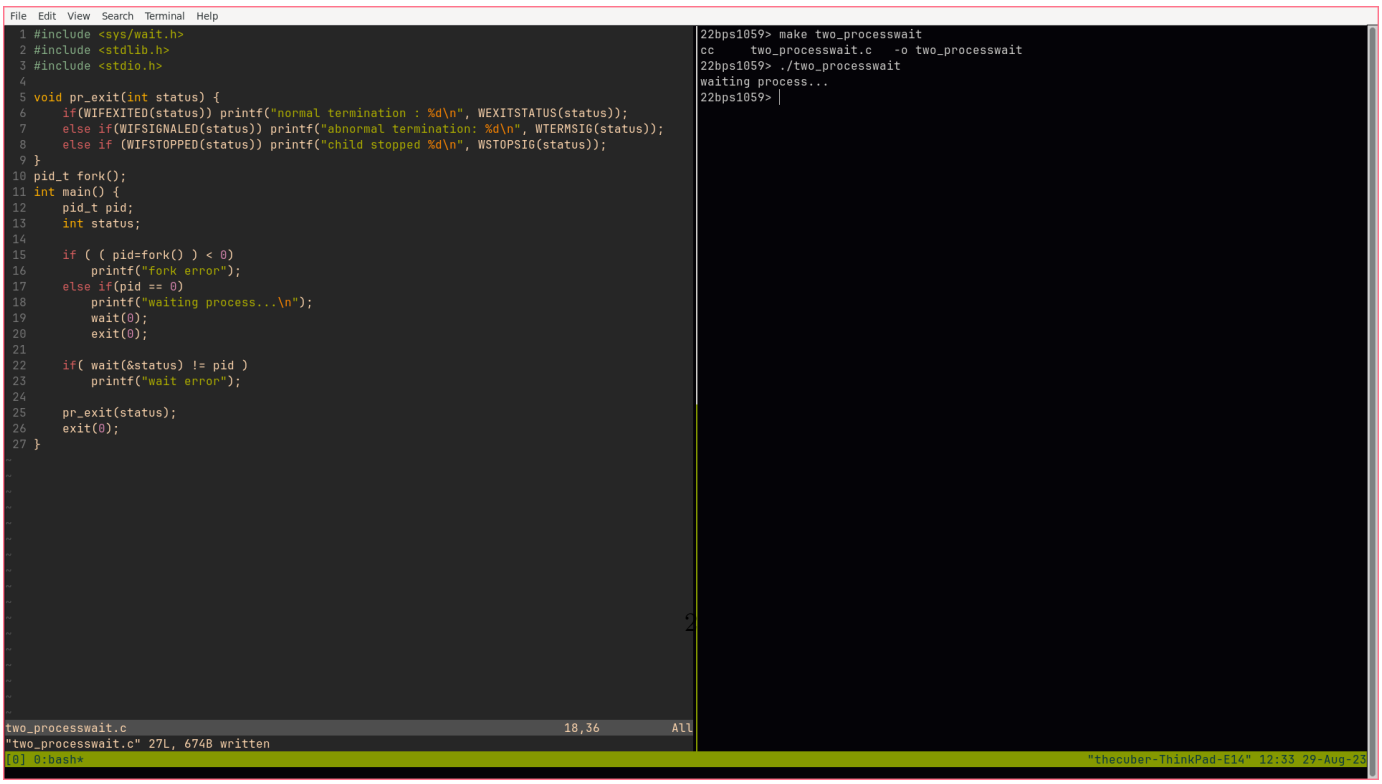
```c
#include <sys/wait.h>
#include <stdlib.h>
#include <stdio.h>

void pr_exit(int status) {
    if(WIFEXITED(status)) printf("normal termination : %d\n", WEXITSTATUS(status));
    else if(WIFSIGNALED(status)) printf("abnormal termination: %d\n", WTERMSIG(status));
    else if (WIFSTOPPED(status)) printf("child stopped %d\n", WSTOPSIG(status));
}
pid_t fork();
int main() {
    pid_t pid;
    int status;

    // process 1 (just exits)
    if ( ( pid=fork() ) < 0)
        printf("fork error");
    else if(pid == 0)
        exit(0);

    if( wait(&status) != pid )
        printf("wait error");

    pr_exit(status);

    // process 2 (aborts)
    if( (pid=fork() ) < 0)
        printf("fork error");
    else if (pid == 0)
        abort();

    if( wait(&status) != pid ) printf("wait error");

    pr_exit(status);


    exit(0);
}
```

```
one_processExit.c                                              1,1           All
```

```
22bps1059> ./one_processExit
normal termination : 0
abnormal termination: 6
22bps1059> scrot
22bps1059>
```

```
[0] 0:bash*                                                    "thecuber-ThinkPad-E14" 12:30 29-Aug-23
```

second



```
File  Edit  View  Search  Terminal  Help
```

```c
#include <sys/wait.h>
#include <stdlib.h>
#include <stdio.h>

void pr_exit(int status) {
    if(WIFEXITED(status)) printf("normal termination : %d\n", WEXITSTATUS(status));
    else if(WIFSIGNALED(status)) printf("abnormal termination: %d\n", WTERMSIG(status));
    else if (WIFSTOPPED(status)) printf("child stopped %d\n", WSTOPSIG(status));
}
pid_t fork();
int main() {
    pid_t pid;
    int status;

    if ( ( pid=fork() ) < 0)
        printf("fork error");
    else if(pid == 0)
        printf("waiting process...\n");
        wait(0);
        exit(0);

    if( wait(&status) != pid )
        printf("wait error");

    pr_exit(status);
    exit(0);
}
```

```
two_processwait.c                                              18,36          All
"two_processwait.c" 27L, 674B written
```

```
22bps1059> make two_processwait
cc     two_processwait.c   -o two_processwait
22bps1059> ./two_processwait
waiting process...
22bps1059>
```

```
[0] 0:bash*                                                    "thecuber-ThinkPad-E14" 12:33 29-Aug-23
```

third

File Edit View Search Terminal Help

```c
#include <sys/wait.h>
#include <stdlib.h>
#include <stdio.h>
void pr_exit(int status) {
    if(WIFEXITED(status)) printf("normal termination : %d\n", WEXITSTATUS(status));
    else if(WIFSIGNALED(status)) printf("abnormal termination: %d\n", WTERMSIG(status));
    else if (WIFSTOPPED(status)) printf("child stopped %d\n", WSTOPSIG(status));
}
pid_t fork(); int execvp(); int sleep(int);
int main() {
    pid_t pid;int status;
    // process 1
    if ((pid = fork()) < 0) printf("fork error");
    else if(pid == 0){
        char *command = "cat";
        char *args[] = {command,"file.txt", NULL};
        execvp(command, args);
        exit(0);
    }
    if(wait(&status) != pid) printf("wait error");
    pr_exit(status);
    // process 2
    if((pid = fork()) < 0) printf("fork error");
    else if (pid == 0) {
        abort();
    }
    if(wait(&status) != pid) printf("wait error");
    pr_exit(status);
    // process 3
    if((pid = fork()) < 0) printf("fork error");
    else if (pid == 0){
        exit(0);
    }
    if(wait(&status) != pid) printf("wait error");
    pr_exit(status);
    // process 4
    if((pid = fork()) < 0) printf("fork error");
    else if (pid == 0){
        sleep(2);
        exit(0);
    }
    if(wait(&status) != pid) printf("wait error");
    pr_exit(status);
    exit(0);
```

three_entirelifecycle.c 22,5 Top

"three_entirelifecycle.c" 46L, 1318B written

```
22bps1059> make three_entirelifecycle
cc     three_entirelifecycle.c   -o three_entirelifecycle
22bps1059> ./three_entirelifecycle
this is some file
with text in it
normal termination : 0
abnormal termination: 6
normal termination : 0
normal termination : 0
22bps1059>
```

[0] 0:bash* "thecuber-ThinkPad-E14" 12:35 29-Aug-23

fourth

File Edit View Search Terminal Help

```c
#include <sys/wait.h>
#include <signal.h>
#include <stdlib.h>
#include <stdio.h>

void pr_exit(int status) {
    if(WIFEXITED(status)) printf("normal termination : %d\n", WEXITSTATUS(status));
    else if(WIFSIGNALED(status)) printf("abnormal termination: %d\n", WTERMSIG(status));
    else if (WIFSTOPPED(status)) printf("child stopped %d\n", WSTOPSIG(status));
}

int getpid(); int getppid(); int fork(); int sleep(); int pause(); int kill();

int main() {
    int processTimes[] = {7,1,2,1,5};
    int numofprocesses = sizeof(processTimes) / sizeof(processTimes[0]);
    pid_t processpids[numofprocesses]; /* W: variable 'processpids' set but not used [-Wun

    //initializing processes
    for(int i = 0; i<numofprocesses; i++) {
        pid_t pid = fork();
        int status;
        processpids[i] = pid;

        if (pid < 0)
            perror("fork error");

        else if(pid == 0){
        printf("\nim process %i my pid is %d and my parent is %d\n",i,getpid(),getppid());
            fflush(stdout);
            sleep(processTimes[i]);
            exit(0);
        }

        if( wait(&status) != pid )
            printf("wait error\n");

        pr_exit(status);
    }

    exit(0);
    return 0;
}
```

fcfs.c 11,0-1 All

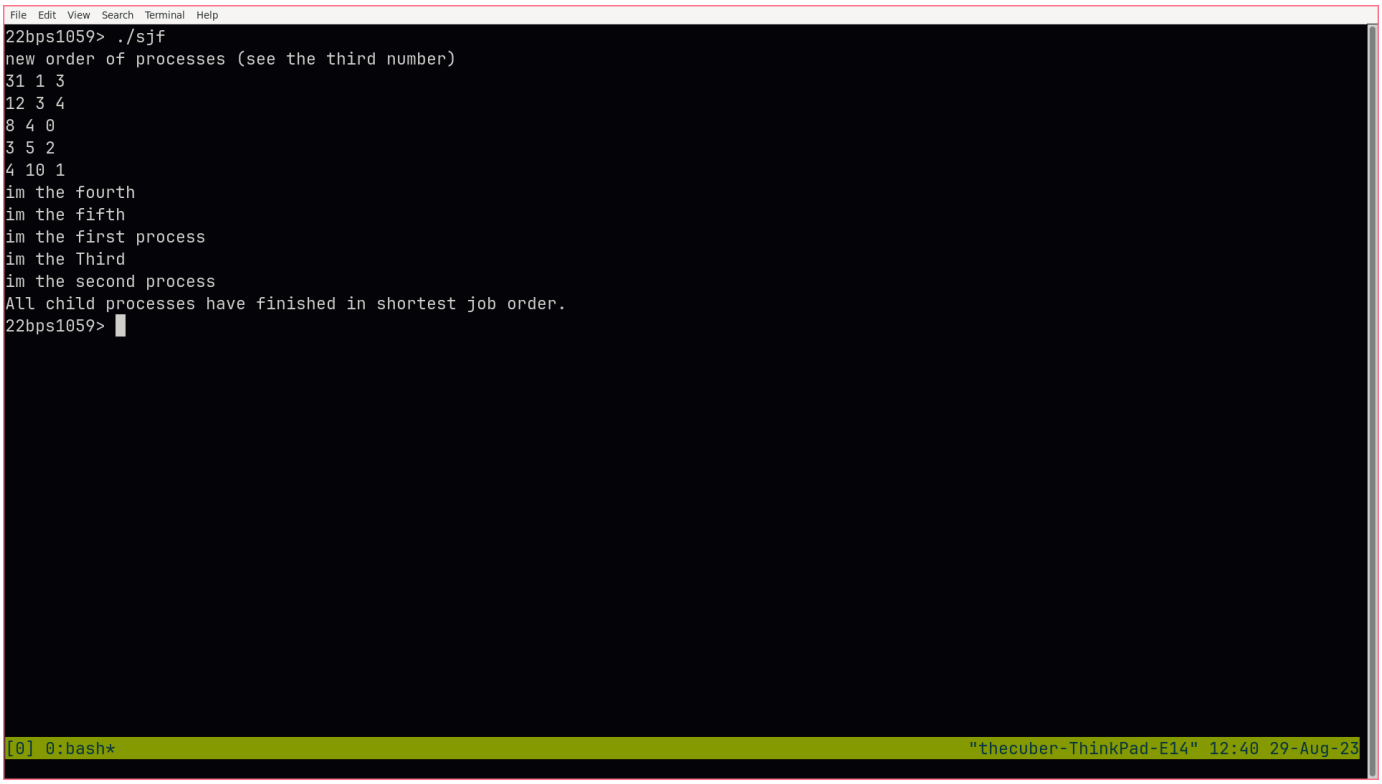
```
22bps1059> make fcfs
cc     fcfs.c   -o fcfs
22bps1059> ./fcfs

im process 0 my pid is 235973 and my parent is 235972
normal termination : 0

im process 1 my pid is 235974 and my parent is 235972
normal termination : 0

im process 2 my pid is 235975 and my parent is 235972
normal termination : 0

im process 3 my pid is 235976 and my parent is 235972
normal termination : 0

im process 4 my pid is 235977 and my parent is 235972
normal termination : 0
22bps1059>
```

[0] 0:bash* "thecuber-ThinkPad-E14" 12:36 29-Aug-23

fifth

```c
///22bps1059
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>

void swap( int *x, int *y){
    int c = *x;
    *x = *y;
    *y = c;
}
int main() {
    // array of arrival time, bursttime, processnumber
    int processtable[5][3] = { {8,4,0},{4,10,1},{3,5,2},{31,1,3},{12,3,4} };
    int numprocesses = 5;

    // sorting accoring to bursttime to get sjfirst
    for(int i = 0; i< numprocesses; i++) {
        for(int j = 0; j<numprocesses-1; j++) {
            if(processtable[j][1] > processtable[j+1][1]) {
                swap( &processtable[j][0],&processtable[j+1][0] );
                swap( &processtable[j][1],&processtable[j+1][1] );
                swap( &processtable[j][2],&processtable[j+1][2] );
            }
        }
    }

    printf("new order of processes (see the third number)\n");
    for(int i = 0; i< numprocesses; i++) {
        for(int j = 0; j<3; j++) {
            printf("%d ",processtable[i][j]);
        }printf("\n");
    }
```

```
    }

    for(int i = 0; i<numprocesses; i++) {
        int processdecider = processtable[i][2];
        pid_t pid = fork(); int  status;
        if (pid == -1) {
            perror("fork error");
            exit(1);
        } else if (pid == 0) {
            sleep(processtable[i][1]);
            switch(processdecider){
                case 0:
                    fflush(stdout);
                    printf("im the first process\n");
                    break;

                case 1:
                    fflush(stdout);
                    printf("im the second process\n");
                    break;

                case 2:
                    fflush(stdout);
                    printf("im the Third\n");
                    break;

                case 3:
                    fflush(stdout);
                    printf("im the fourth\n");
                    break;

                case 4:
                    printf("im the fifth\n");
                    break;
            }
            exit(0);
        }
        if( wait(&status) != pid )
            printf("wait error\n");
    }

    // Wait for child processes using
    while(wait(NULL) > 0);

    printf("All child processes have finished in shortest job order.\n");
    return 0;
}
```


File Edit View Search Terminal Help
22bps1059> ./sjf
new order of processes (see the third number)
31 1 3
12 3 4
8 4 0
3 5 2
4 10 1
im the fourth
im the fifth
im the first process
im the Third
im the second process
All child processes have finished in shortest job order.
22bps1059>
[0] 0:bash*
"thecuber-ThinkPad-E14" 12:40 29-Aug-23

22bps1059

$lab - 3$

first

```c
#include <stdio.h>

int main() {
    fork();  /* W: implicit declaration of function 'fork' [-W
    printf("%d\n", getpid()); /* W: implicit declaration of fu
}
```

```
~/coding/os/22bps1059 > ./one
116062
116064
~/coding/os/22bps1059 >
```

second

```c
#include <stdio.h>

int main() {
    int a = 14; int b = 3;
    int p1 = fork();  /* W: implicit declaration of function '
    int p2 = fork();
    if(!p1 && p2) {
        printf("this is %d, child of %d\n", getpid(), getppid(
)); /* W: implicit declaration of function 'getppid' [-Wimplic
        printf("sum of %d and %d is %d\n",a,b,a+b);
    }
    else if (p1 && !p2) {
        printf("this is %d, child of %d\n", getpid(), getppid(
));
        printf("product of %d and %d is %d\n",a,b,a*b);
    }
    printf("\n");
}
```

```
~/coding/os/22bps1059 > ./one

this is 198009, child of 198006
product of 14 and 3 is 42


this is 198008, child of 198006
sum of 14 and 3 is 17

~/coding/os/22bps1059 >
```

third

```c
#include <stdio.h>

int main() {
    char *command = "cat";
    char *args[] = {command,"file.txt",NULL};
    int pid = fork(); /* W: implicit declaration of function '
    if(pid>0)
        execvp(command, args); /* W: implicit declaration of f
    return 0;
}
```

```
~/coding/os/22bps1059 > ./two
This is a file
with several lines
of text in it
~/coding/os/22bps1059 >
```

fourth

```c
#include <stdio.h>
int main() {
    int p1 = fork();  /* W: implicit declaration of function '
    int p2 = fork();
    if(!p1 && p2) {
        printf("Writing to filename.txt\n");
        char *command = "./threebin";
        char *args[] = {command,NULL};
        execvp(command, args); /* W: implicit declaration of f
    }
    else if (p1 && !p2) {
        char *command = "cat";
        char *args[] = {command,"urregno.txt",NULL};
        execvp(command, args);
    } return 0;
}
```

```c
#include <stdio.h>
int main() {
    FILE *file = fopen("urregno.txt","w");
    if(file==NULL) {
        printf("failed to open file\n");
        return 1;
    }
    fprintf(file,"22BPS1059!\n");
    fclose(file);
    printf("written to file \n");
    return 0;
}
```

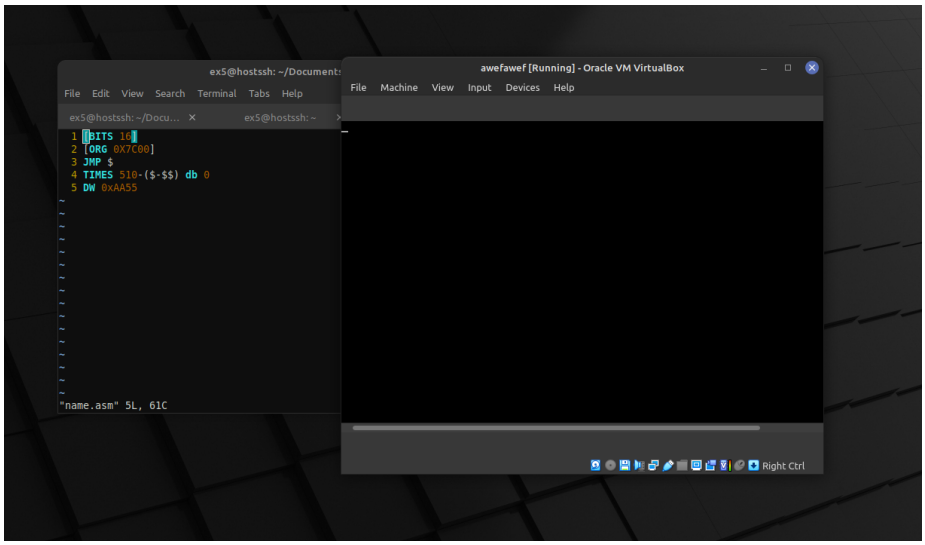
```
threebin.c
```

```
~/coding/os/22bps1059 > ./three                08/1
Writing to filename.txt
written to file
22BPS1059!
```

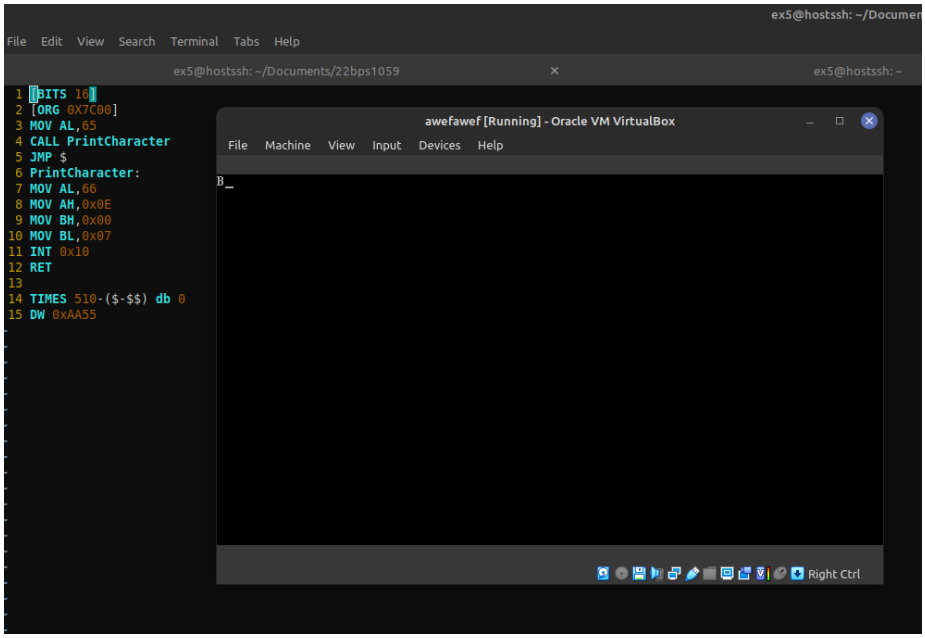
22bps1059

$lab - 2$

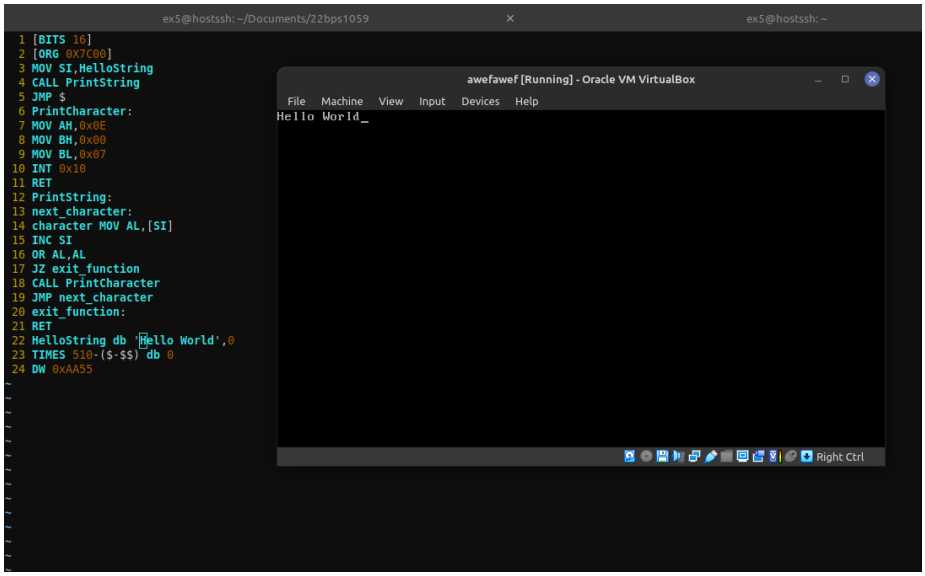
first



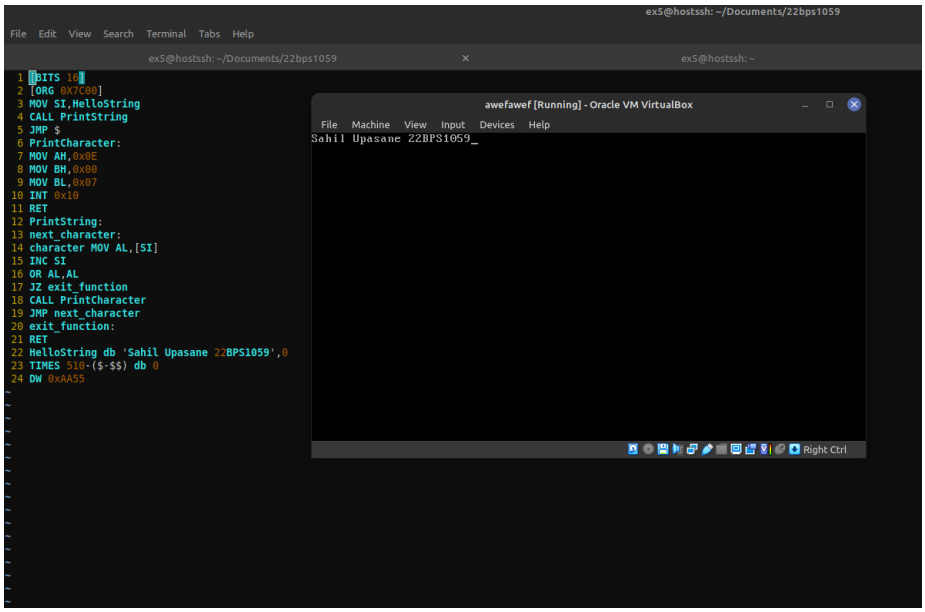
second



third



fourth



$22bps1059$

```
cardi~/Documents/vit/os/22bps1059 date
Wednesday 26 July 2023 07:44:16 PM IST
cardi~/Documents/vit/os/22bps1059 name="sahil"
cardi~/Documents/vit/os/22bps1059 echo $name
sahil
cardi~/Documents/vit/os/22bps1059 echo $$
410003
```

```
cardi~/Documents/vit/os/22bps1059 date
Wednesday 26 July 2023 07:44:16 PM IST
cardi~/Documents/vit/os/22bps1059 name="sahil"
cardi~/Documents/vit/os/22bps1059 echo $name
sahil
cardi~/Documents/vit/os/22bps1059 echo $$
410003
cardi~/Documents/vit/os/22bps1059 ls
cardi~/Documents/vit/os/22bps1059 cd ..
cardi~/Documents/vit/os ls
22bps1059  boat.png  ostemplate.tex  texput.log
cardi~/Documents/vit/os ls -l
total 84
drwxrwxr-x 2 thecuber thecuber  4096 Jul 26 19:00 22bps1059
-rw-rw-r-- 1 thecuber thecuber 72254 Mar  8  2021 boat.png
-rw-rw-r-- 1 thecuber thecuber  1877 Jul 26 18:56 ostemplate.tex
-rw-rw-r-- 1 thecuber thecuber   675 Jul 26 14:09 texput.log
cardi~/Documents/vit/os ls -a
.  ..  22bps1059  boat.png  ostemplate.tex  texput.log
cardi~/Documents/vit/os cat ostemplate.tex
% This is a small sample LaTeX input file (Version of 10 April 1994)
%
```

```
  1 this is text in the file
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
trial.moon [+]                                  1,25          All
-- INSERT --
```

```
cardi~/Documents/vit/os cd 22bps1059/
cardi~/Documents/vit/os/22bps1059 vi trial.moon
cardi~/Documents/vit/os/22bps1059 cat trial.moon
this is text in the file
cardi~/Documents/vit/os/22bps1059 cp trial.moon trialcopy.moon
cardi~/Documents/vit/os/22bps1059 ls
trialcopy.moon  trial.moon
cardi~/Documents/vit/os/22bps1059 rm trial.moon
cardi~/Documents/vit/os/22bps1059 mkdir asfd
cardi~/Documents/vit/os/22bps1059 ls
asfd  trialcopy.moon
cardi~/Documents/vit/os/22bps1059 rmdir asdf
rmdir: failed to remove 'asdf': No such file or directory
cardi~/Documents/vit/os/22bps1059 ls
asfd  trialcopy.moon
cardi~/Documents/vit/os/22bps1059 rmdir asfd
cardi~/Documents/vit/os/22bps1059
```

```
cardi~/Documents/vit/os/22bps1059 chmod 777 trialcopy.moon
cardi~/Documents/vit/os/22bps1059 ps
    PID TTY          TIME CMD
 410003 pts/3    00:00:00 bash
 410373 pts/3    00:00:00 ps
cardi~/Documents/vit/os/22bps1059 ps -f
UID          PID    PPID  C STIME TTY          TIME CMD
thecuber  410003  400032  0 19:40 pts/3    00:00:00 bash
thecuber  410374  410003  0 19:53 pts/3    00:00:00 ps -f
cardi~/Documents/vit/os/22bps1059
```

```
top - 19:53:36 up 2 days, 15 min,  1 user,  load average: 1.56, 1.66, 1.73
Tasks: 297 total,   2 running, 295 sleeping,   0 stopped,   0 zombie
%Cpu(s): 12.0 us,  4.2 sy,  0.0 ni, 83.8 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   7616.0 total,    694.7 free,   3882.8 used,   3038.5 buff/cache
MiB Swap:      0.0 total,      0.0 free,      0.0 used.   2626.6 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
  11754 thecuber  20   0   13.2g 715256 155164 S  41.2   9.2 197:07.29 firefox
 262534 thecuber  20   0 3139124 525816 106876 R  35.3   6.7  83:34.03 Isolated Web Co
   3160 thecuber  20   0 6767464 308976  84784 S  11.8   4.0  63:50.95 gnome-shell
  63764 thecuber  20   0  546048  64276  35940 S  11.8   0.8  24:21.57 RDD Process
 410391 thecuber  20   0   21892   4284   3424 R   5.9   0.1   0:00.03 top
      1 root      20   0  167008   9984   5924 S   0.0   0.1   2:45.09 systemd
      2 root      20   0       0      0      0 S   0.0   0.0   0:00.11 kthreadd
      3 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_gp
      4 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 rcu_par_gp
      5 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 slub_flushwq
      6 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 netns
     10 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 mm_percpu_wq
     11 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_kthread
     12 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_rude_kthread
     13 root      20   0       0      0      0 I   0.0   0.0   0:00.00 rcu_tasks_trace_kthread
     14 root      20   0       0      0      0 S   0.0   0.0   0:02.69 ksoftirqd/0
     15 root      20   0       0      0      0 I   0.0   0.0   1:05.41 rcu_preempt
     16 root      rt   0       0      0      0 S   0.0   0.0   0:00.42 migration/0
     17 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/0
     19 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
     20 root      20   0       0      0      0 S   0.0   0.0   0:00.03 cpuhp/1
     21 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/1
     22 root      rt   0       0      0      0 S   0.0   0.0   0:00.50 migration/1
     23 root      20   0       0      0      0 S   0.0   0.0   0:01.89 ksoftirqd/1
     25 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/1:0H-events_highpri
     26 root      20   0       0      0      0 S   0.0   0.0   0:00.01 cpuhp/2
     27 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/2
     28 root      rt   0       0      0      0 S   0.0   0.0   0:00.54 migration/2
     29 root      20   0       0      0      0 S   0.0   0.0   0:01.24 ksoftirqd/2
     31 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 kworker/2:0H-events_highpri
     32 root      20   0       0      0      0 S   0.0   0.0   0:00.00 cpuhp/3
     33 root     -51   0       0      0      0 S   0.0   0.0   0:00.00 idle_inject/3
     34 root      rt   0       0      0      0 S   0.0   0.0   0:00.53 migration/3
     35 root      20   0       0      0      0 S   0.0   0.0   0:01.17 ksoftirqd/3
```

```
cardi~/Documents/vit/os/22bps1059 ls | grep trial
trialcopy.moon
cardi~/Documents/vit/os/22bps1059 ifconfig
Command 'ifconfig' not found, but can be installed with:
sudo apt install net-tools
cardi~/Documents/vit/os/22bps1059 sudo apt install net-tools
[sudo] password for thecuber:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  gir1.2-ayatanaappindicator3-0.1
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 105 not upgraded.
Need to get 204 kB of archives.
After this operation, 819 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 net-tools amd64 1.60+git20181103.0eebece-1ubuntu5 [204 kB]
Fetched 204 kB in 2s (83.8 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 255791 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-1ubuntu5_amd64.deb ...
Unpacking net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Setting up net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Processing triggers for man-db (2.10.2-1) ...
cardi~/Documents/vit/os/22bps1059 ifconfig
enp4s0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 54:05:db:a9:41:a1  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 97046  bytes 13293939 (13.2 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 97046  bytes 13293939 (13.2 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

```
cardi~/Documents/vit/os/22bps1059 ping google.com
PING google.com (142.250.77.174) 56(84) bytes of data.
64 bytes from maa05s17-in-f14.1e100.net (142.250.77.174): icmp_seq=1 ttl=57 time=3.62 ms
64 bytes from maa05s17-in-f14.1e100.net (142.250.77.174): icmp_seq=2 ttl=57 time=5.28 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 3.623/4.451/5.279/0.828 ms
cardi~/Documents/vit/os/22bps1059 ls
trialcopy.moon
cardi~/Documents/vit/os/22bps1059 scp trialcopy.moon trial.moon
cardi~/Documents/vit/os/22bps1059 ls
trialcopy.moon  trial.moon
cardi~/Documents/vit/os/22bps1059
```

second

```
cardi~/Documents/vit/os$ cd 22bps1059/
cardi~/Documents/vit/os/22bps1059$ ls
trialcopy.moon  trial.moon
cardi~/Documents/vit/os/22bps1059$ mkdir 22BPS1059
cardi~/Documents/vit/os/22bps1059$ mv trial.moon 22BPS1059/trial.moon
cardi~/Documents/vit/os/22bps1059$ ls
22BPS1059  trialcopy.moon
cardi~/Documents/vit/os/22bps1059$ cd 22BPS1059/
cardi~/Documents/vit/os/22bps1059/22BPS1059$ ls
trial.moon
cardi~/Documents/vit/os/22bps1059/22BPS1059$ cd ..
cardi~/Documents/vit/os/22bps1059$ ls
22BPS1059  trialcopy.moon
cardi~/Documents/vit/os/22bps1059$
```

third

```
1 #!/bin/bash
2
3
4 touch filename.txt
5 ls
6 mkdir 22bps1059a
7 ls
8 cd 22bps1059a
9 ls
10 cd ..
11 mv filename.txt 22bps1059a/filename.txt
12 cd 22bps1059a
13 ls
```

```
cardi~/Documents/vit/os/22bps1059$ bash script.sh
22BPS1059  filename.txt  script.sh  trialcopy.moon
22BPS1059  22bps1059a  filename.txt  script.sh  trialcopy.moon
filename.txt
```