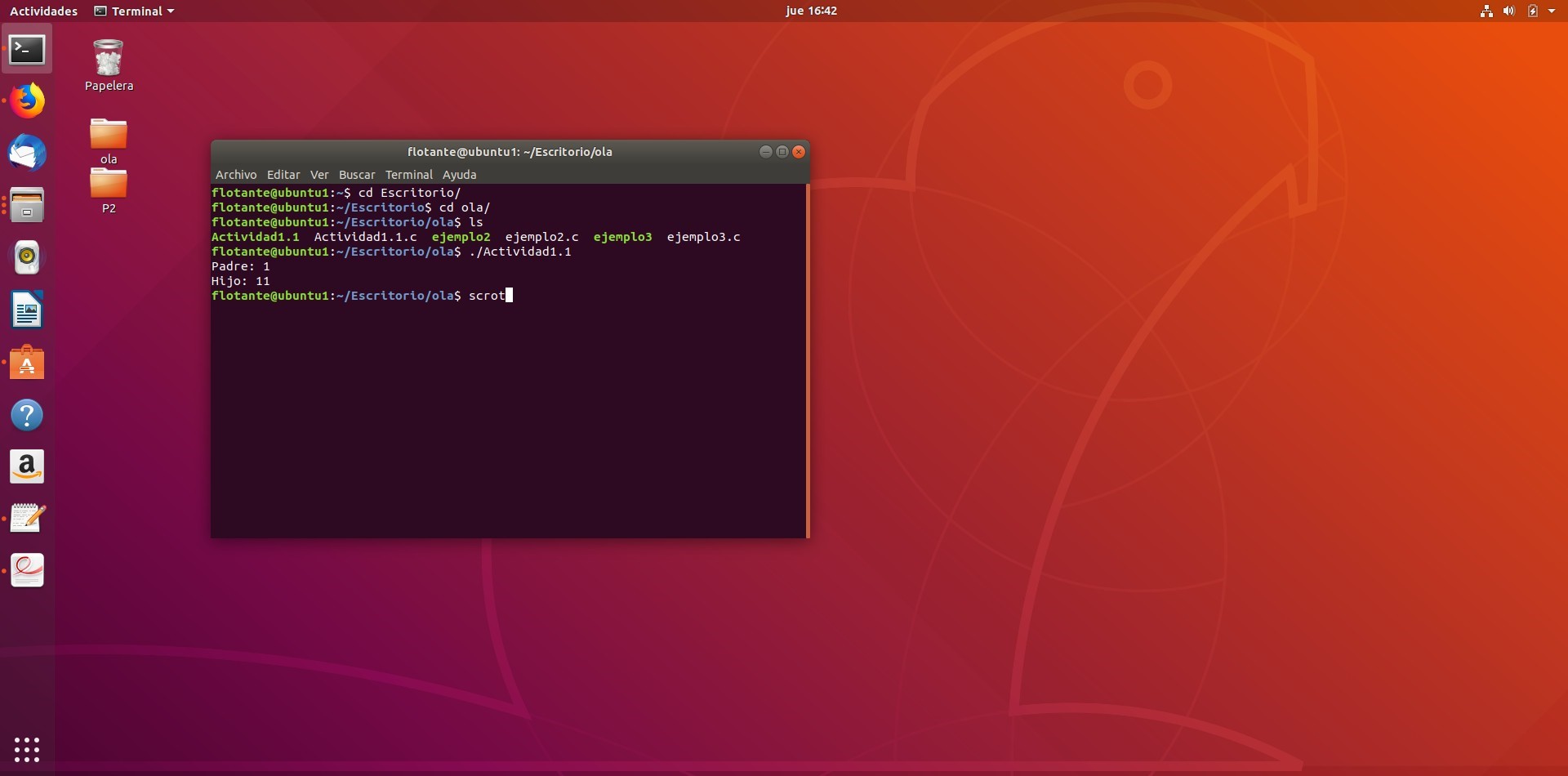
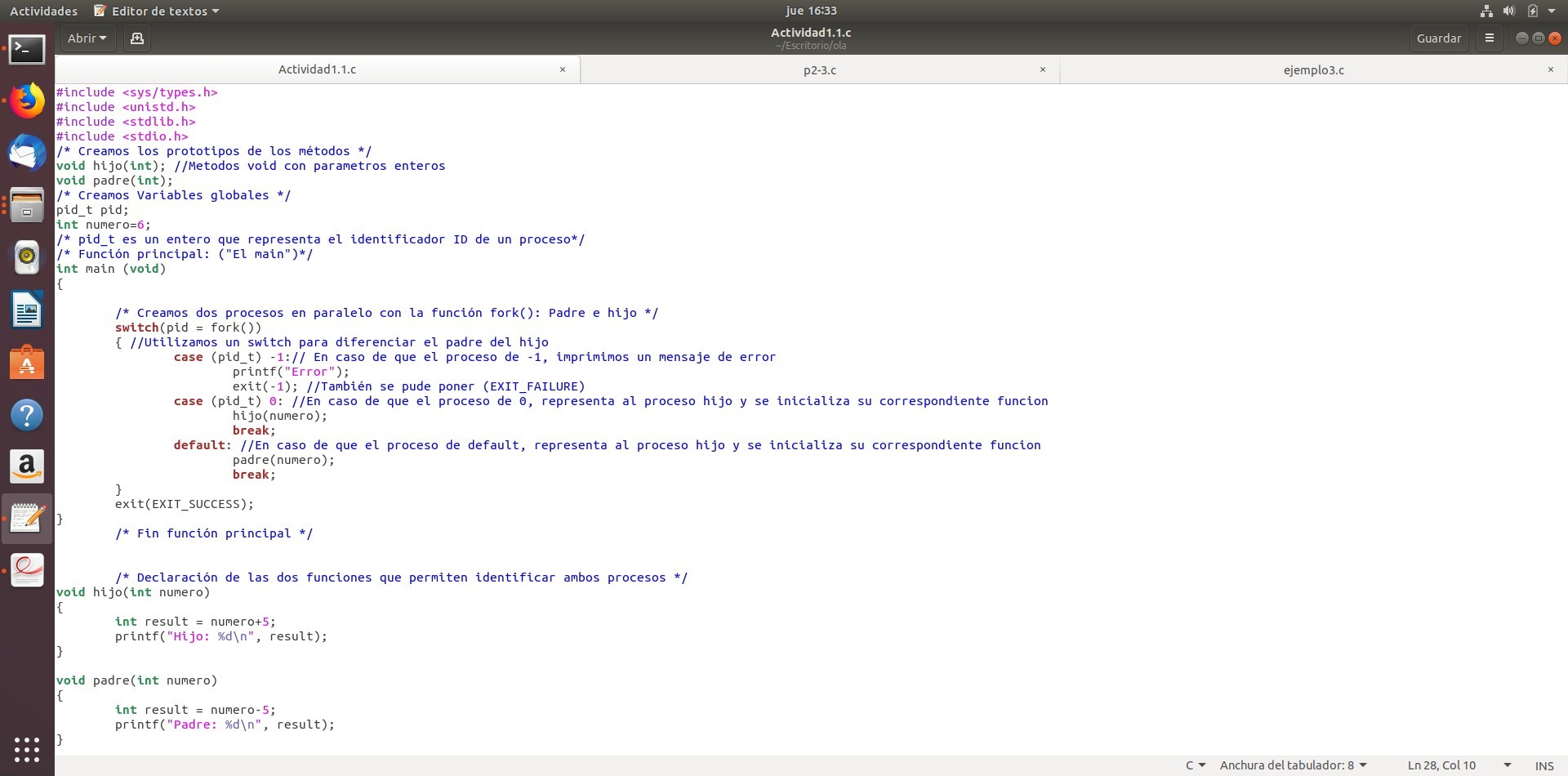
Actividad 1.1

Esto es el terminal:



Esto es el código:



Realiza un programa en C que cree un proceso (tendremos 2 procesos uno padre y otro hijo). El programa definirá una variable entera y le pasaremos el valor 6. El proceso padre incrementará dicho valor en 5 y el hijo restará 5. Se deben mostrar los valores en pantalla.

Código:

#include <sys/types.h>

#include <unistd.h>

#include <stdlib.h>

#include <stdio.h>

/\* Creamos los prototipos de los métodos \*/

void hijo(int); //Metodos void con parametros enteros

void padre(int);

/\* Creamos Variables globales \*/

pid\_t pid;

int numero=6;

/\* pid\_t es un entero que representa el identificador ID de un proceso\*/

/\* Función principal: ("El main")\*/

int main (void)

{

/\* Creamos dos procesos en paralelo con la función fork(): Padre e hijo \*/

switch(pid = fork())

{ //Utilizamos un switch para diferenciar el padre del hijo

case (pid\_t) -1:// En caso de que el proceso de -1, imprimimos un mensaje de error

printf("Error");

exit(-1); //También se pude poner (EXIT\_FAILURE)

case (pid\_t) 0: //En caso de que el proceso de 0, representa al proceso hijo y se inicializa su correspondiente funcion

hijo(numero);

break;

default: //En caso de que el proceso de default, representa al proceso hijo y se inicializa su correspondiente funcion

padre(numero);

break;

}

exit(EXIT\_SUCCESS);

}

/\* Fin función principal \*/

/\* Declaración de las dos funciones que permiten identificar ambos procesos \*/

void hijo(int numero)

{

int result = numero+5;

printf("Hijo: %d\n", result);

}

void padre(int numero)

{

int result = numero-5;

printf("Padre: %d\n", result);

}