

Sistemas Operativos

Laboratorio

Introducción al entorno de desarrollo

Índice

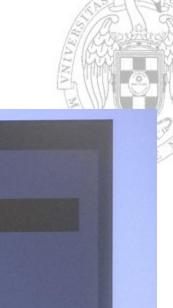
- Primeros pasos GNU/Linux
- El intérprete de comandos Bash
- Páginas de manual
- Proyectos C con Makefile





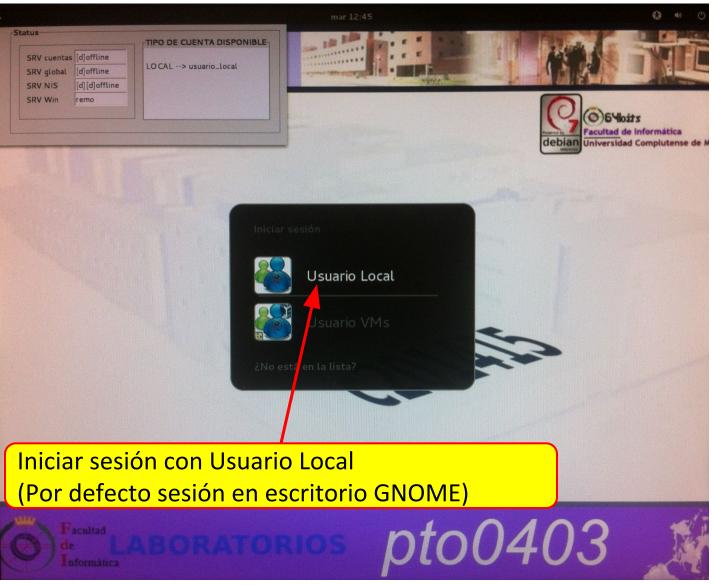
Primeros pasos GNU/Linux

Acceso a instalación nativa: Gestor de arranque





Acceso a instalación nativa: Pantalla de login





Introducción a la máquina virtual

- Se proporciona una imagen de máquina virtual VMWare
 - ✓ Instalar VMWare player (Google it)
 - ✓ Descargar Máquina Virtual de Debian
 - ✓ Abrirla con VMWare (botón derecho → Abrir con VMWare, p.e.)

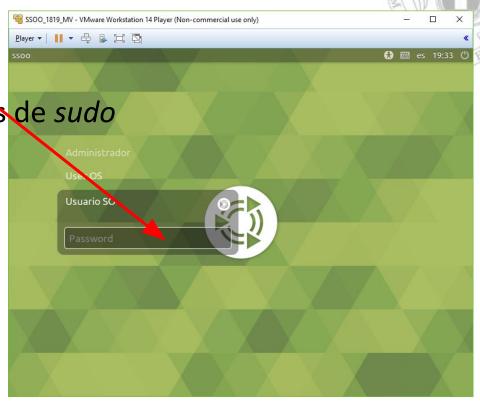


Introducción a la máquina virtual

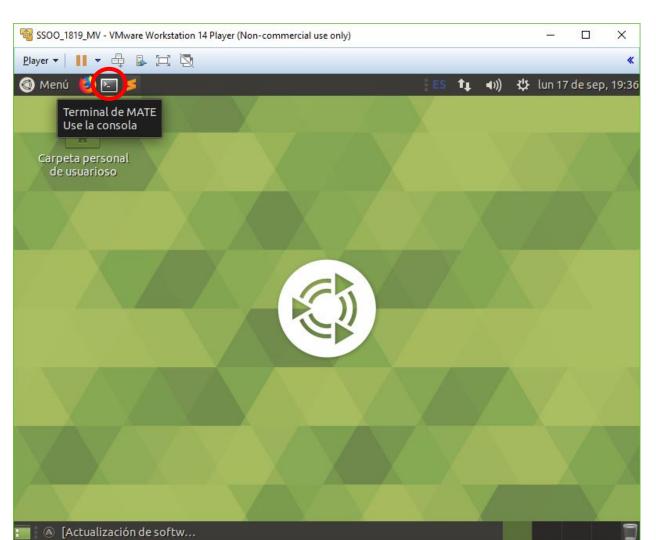
Usuario: usuarioso

Contraseña: usuarioso

Este usuario tiene permisos de sudo

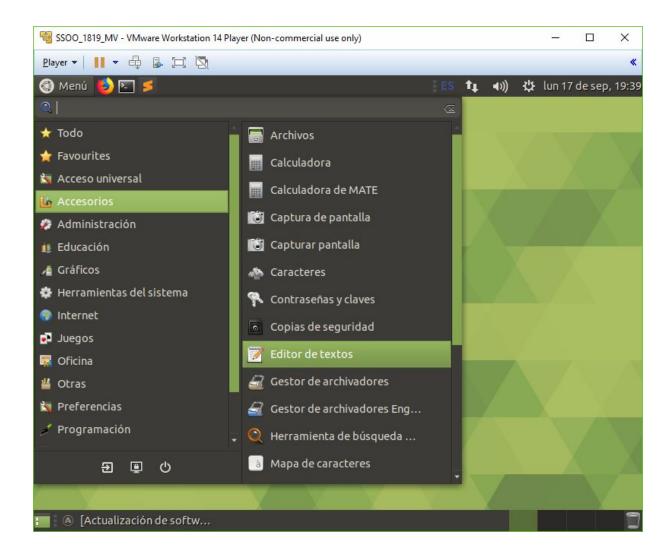


Abrir una ventana de terminal (shell)



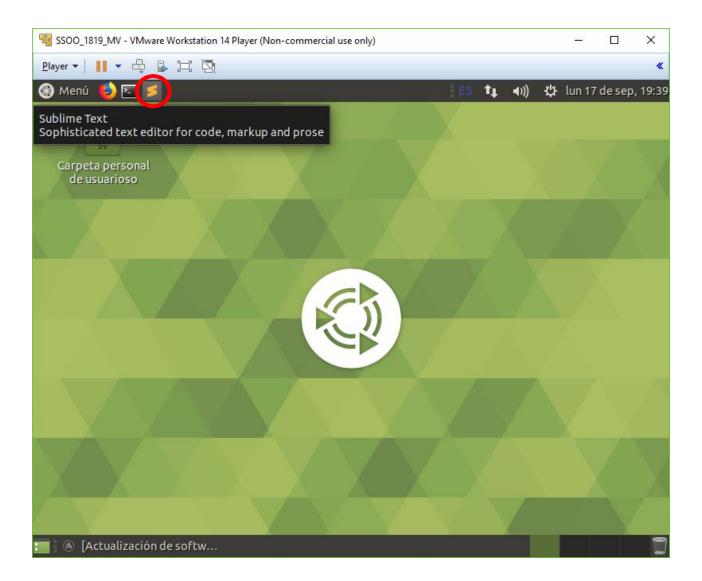


Abrir un editor de textos (gedit)





Abrir otro editor de textos (Sublime Text)



Ejecutando aplicaciones en Linux

Cómo ejecutar aplicaciones:

- Icono de la aplicación
 - Escritorio
 - Menú de aplicaciones del sistema
- Alt+F2 (Introducir comando de la aplicación)
- Desde terminal
 - Abrir una ventana de terminal
 - Ejecutar comando de la aplicación
 - ✓ Sin GUI/ Foreground: <comando>

```
$ Is
$ pwd
```

✓ Con GUI / Background: <comando> &

```
$ gedit &
```

\$ firefox &



11



El intérprete de comandos Bash

Terminal: Comandos básicos

■ Listar el contenido de un directorio (ls):

```
$ Is
$ Is -la
```

Cambiar de directorio (cd):

\$ cd directorioDestino

\$ cd - #Vuelve al directorio anterior

\$ cd #Cambia al \$HOME del usuario

■ Consultar la ruta del directorio de trabajo actual (pwd):

```
$ pwd
```

Copiar un fichero o un directorio (cp)

```
$ cp fichero.txt directorioDestino
```

\$ cp fichero.txt copia_fichero.txt

\$ cp -r directorioOrigen directorioDestino



Terminal: Comandos básicos

Eliminar un fichero o un directorio (rm,rmdir)

\$ rm fichero.txt

\$ rmdir directorio Vacío

\$ rm -rf directorio

Mover/renombrar ficheros o directorios (mv)

\$ mv ficheroViejo.txt ficheroNuevo.txt

\$ mv fichero directorio-destino

\$ mv directorio directorio-destino

Crear un fichero/directorio vacío (touch, mkdir)

\$ touch ficheroVacio.txt

\$ mkdir nuevoDirectorio

Mostrar el contenido de un fichero de texto (cat,more,less)

\$ cat fichero.txt

SO

\$ more fichero.txt

\$ less fichero.txt



14

Terminal: Algunas curiosidades de Bash

- Usar TAB para autocompletar el texto de los comandos o los nombres y rutas de ficheros o directorios
- Posicionar el cursor al principio de la orden (CTRL+A)
- Posicionar el cursor al final de la orden (CTRL+E)
- Historial de comandos
 - ✓ Listar el historial: \$ history
 - ✓ Flechas arriba/abajo
 - ✓ Búsqueda en el historial de comandos (CTRL + R)
- Interrumpir la ejecución de un proceso (CTRL + C)
- Parar temporalmente la ejecución de un proceso en foreground (CTRL + Z)
 - ✓ Reanudar la ejecución de un proceso parado: (fg ó bg)

Terminal: Más información

- Campus Virtual:
 - ✓ Material de apoyo -> "The One Page Linux Manual"
- Manual de referencia de BASH
 - ✓ http://www.gnu.org/software/bash/manual/bashref.html
- Páginas de manual (man <cualquier_comando>)

\$ man cp





Páginas de manual

Páginas de manual en UNIX

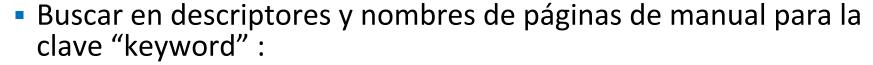
- man da formato y muestra las páginas de manual en línea:
 - Pasa a la siguiente página usando la barra espaciadora
 - Vuelve a la página anterior pulsando la tecla "b"
 - Para salir se pulsa la tecla "q"
- Las páginas de manual están divididas en secciones:
 - (1) Comandos generales
 - (2) Llamadas al sistema
 - (3) Funciones en bibliotecas, especialmente la biblioteca estándar de C
 - (4) Ficheros especiales (generalmente dispositivos, aquellos ubicados en /dev) y controladores (drivers)
 - (5) Formatos de archivo y convenciones
 - (6) Juegos y protectores de pantalla
 - (7) Miscelánea
 - (8) Comandos de administración del sistema y demonios
 - (9) API del kernel (módulos y core kernel)



Uso del programa "man"



```
- man [1-9] command
```



```
- man -k keyword
```

Introducción acerca de una sección

```
- man [1-9] intro
```

Llamadas al sistema Linux :

```
- man 2 syscalls
```

 Buscar texto en la página de manual para los caracteres especificados:

```
- '/caracteres' (siguiente: 'n', anterior: 'N')
```



Subsecciones de una página de manual

- NAME : La primera línea contiene el nombre del comando
- SYNOPSIS: Notación técnica con todas las opciones y/o argumentos que este comando puede aceptar
- DESCRIPTION: Descripción extensa del comando
- OPTIONS: Opciones y su descripción
- ENVIRONMENT: Las variables shell que afectan al comportamiento de este comando
- EXIT STATUS
- BUGS, SEE ALSO, NOTES, EXAMPLES, etc.



Proyectos C con Makefile

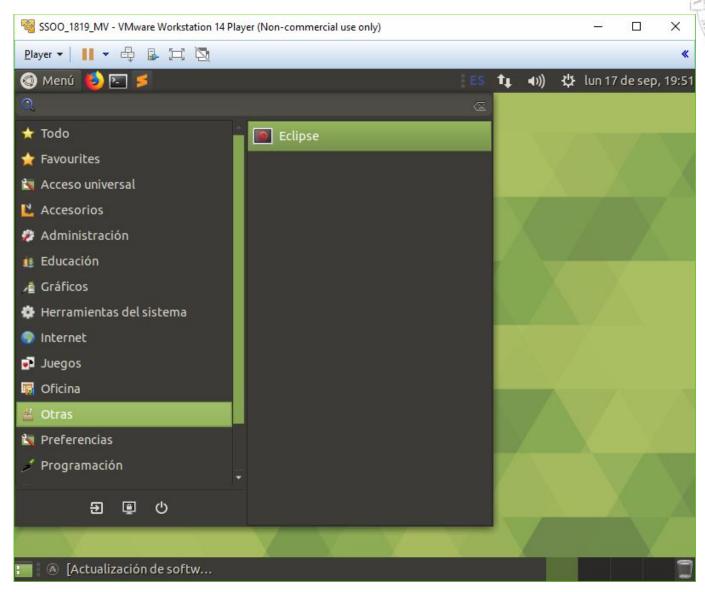
Desarrollo de aplicaciones C en Linux

- Tres alternativas
 - ✓ Editor de Textos + GNU Make + Terminal
 - ✓ Proyecto "C" con Makefile + IDE
 - ✓ Proyecto "C" + IDE (compilación autogestionada)
- Eclipse es el IDE instalado en las distribuciones de GNU/Linux del laboratorio
- Para el desarrollo de las prácticas de esta asignatura usaremos las dos primeras alternativas (basadas en Make)
 - ✓ Con cada práctica y programa de ejemplo se incluirá un Makefile para la compilación del proyecto asociado
 - ✓ Ventaja: según convenga podemos cambiar entre el IDE y el terminal para ejecutar el programa o depurarlo de distintas formas

¿Qué es un Makefile?

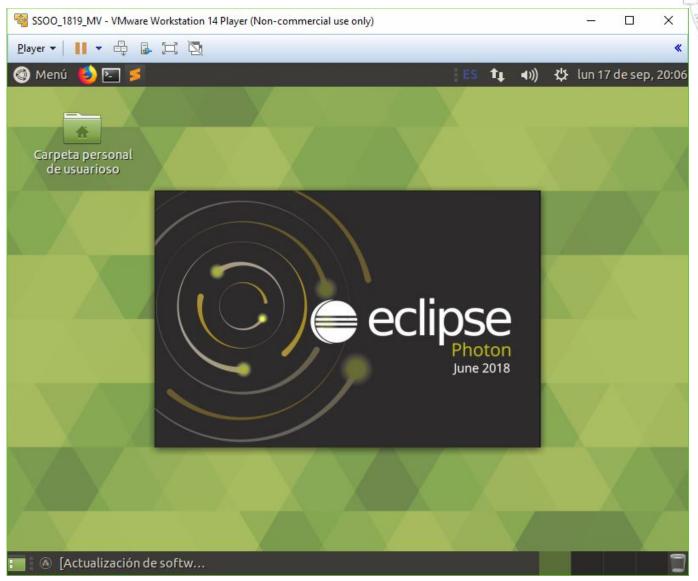
- Fichero de texto que contiene un conjunto de reglas para compilar y enlazar un programa
 - La herramienta make procesa el fichero Makefile que está en el directorio actual y ejecuta los comandos necesarios para construir el ejecutable
 - Hay distintas variantes de make
 - En las prácticas de la asignaturas usaremos GNU make
 - Sintaxis de GNU Make en Sección 1.4.2 del guión de Práctica 1
 - No es necesario que el alumno cree Makefiles
 - Se proporciona uno con cada práctica y código de ejemplo

Ejecución de Eclipse



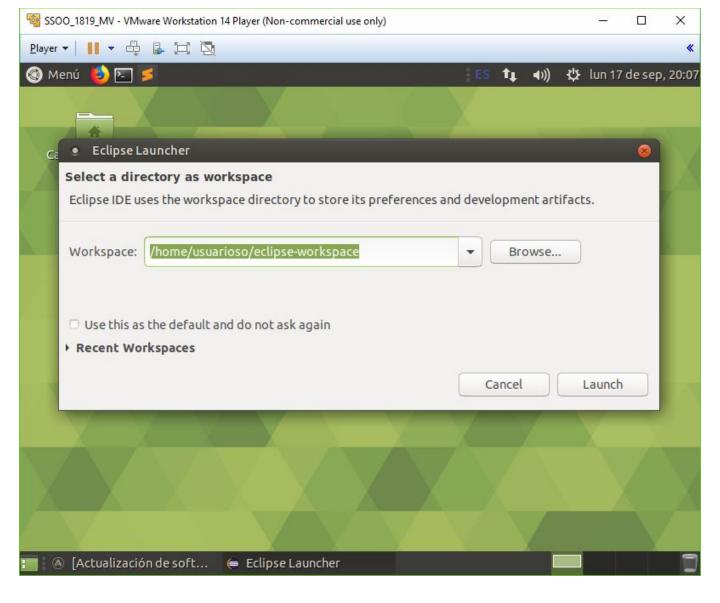


Ejecución de Eclipse (II)



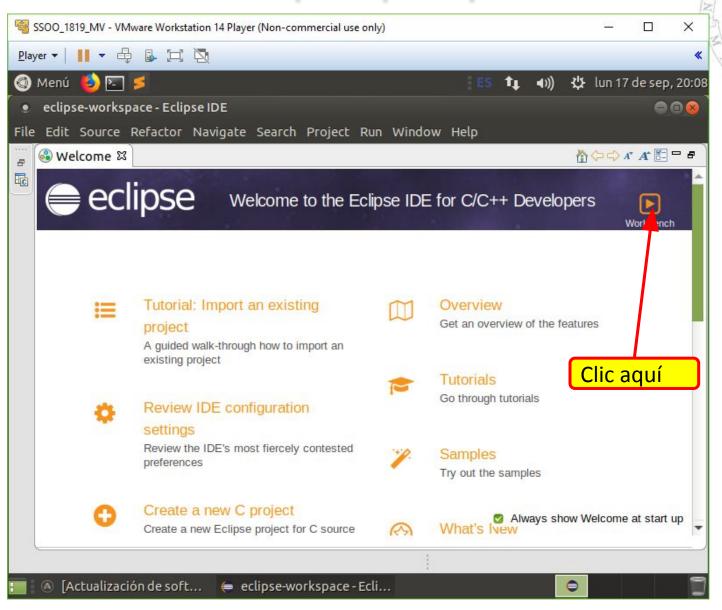


Apertura/creación de workspace



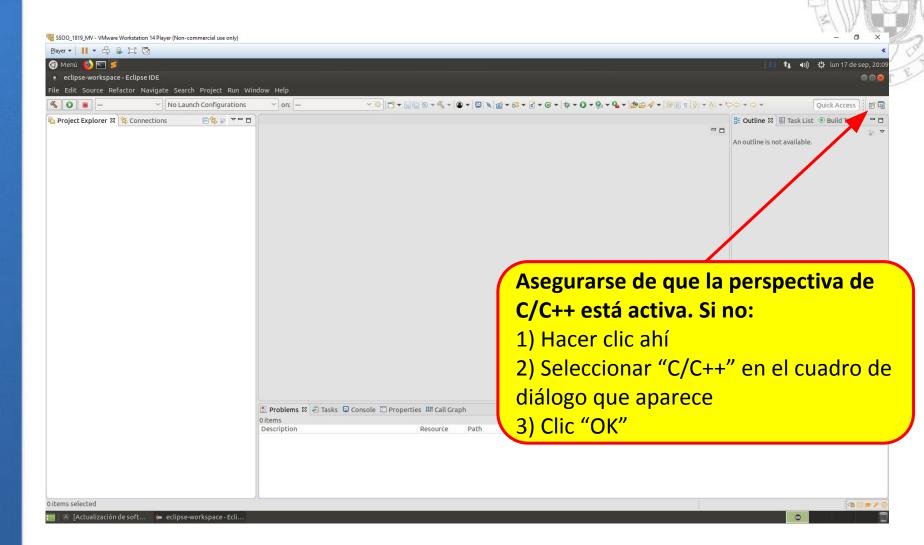


Abriendo eclipse por primera vez



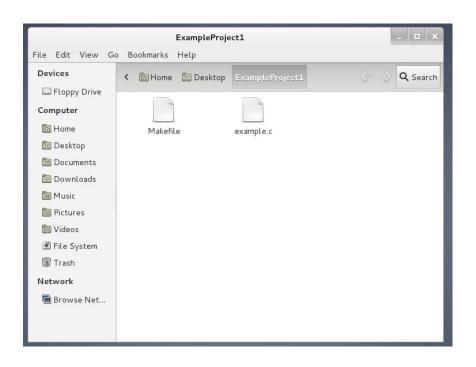


Abriendo eclipse por primera vez



Crear un proyecto con Makefile

Nuestro objetivo es importar un proyecto con Makefile ubicado en ~/Escritorio/ExampleProject1



```
#include <stdio.h>
int main(int argc, char *argv[]) {
  printf("Hi%s %s!\n", argv[0], argv[1]);
  return 0;
}
```

```
all: example

example: example.c

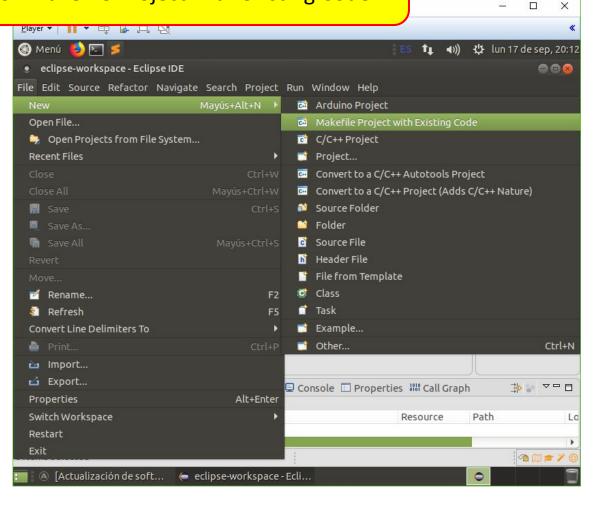
gcc -g -Wall example.c -o example

clean:
-rm –f example
```

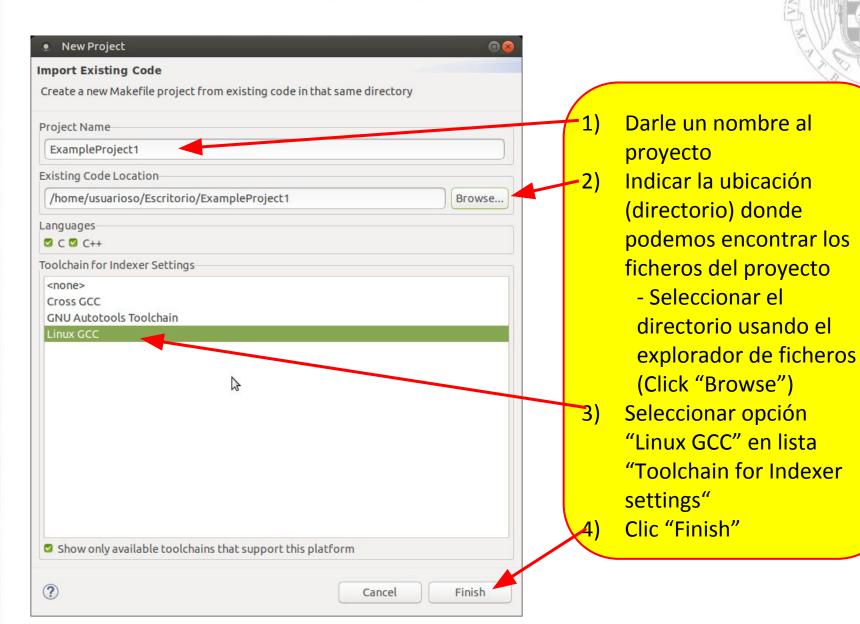
Creando un proyecto con Makefile

Objetivo: importar proyecto con Makefile

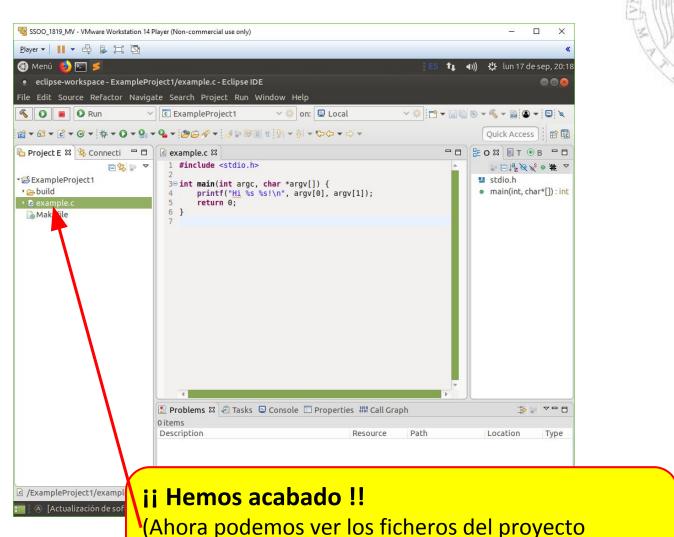
Crear Nuevo "Makefile Project with existing Code"



Creando un proyecto con Makefile



Creando un proyecto con Makefile

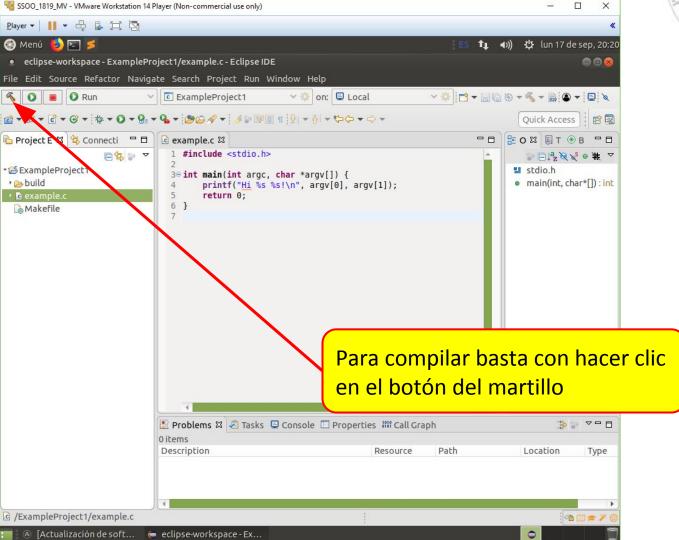


desplegando el menú del explorador de Proyectos y

haciendo clic en el fichero correspondiente)

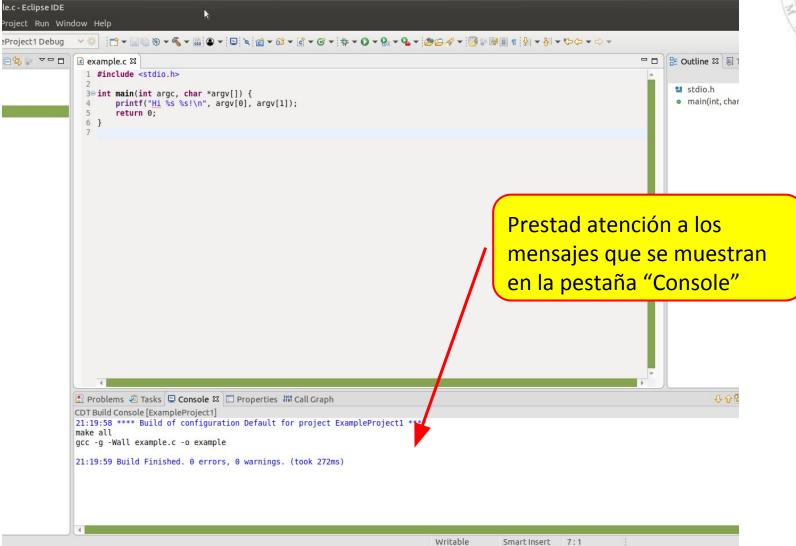


Compilación del proyecto



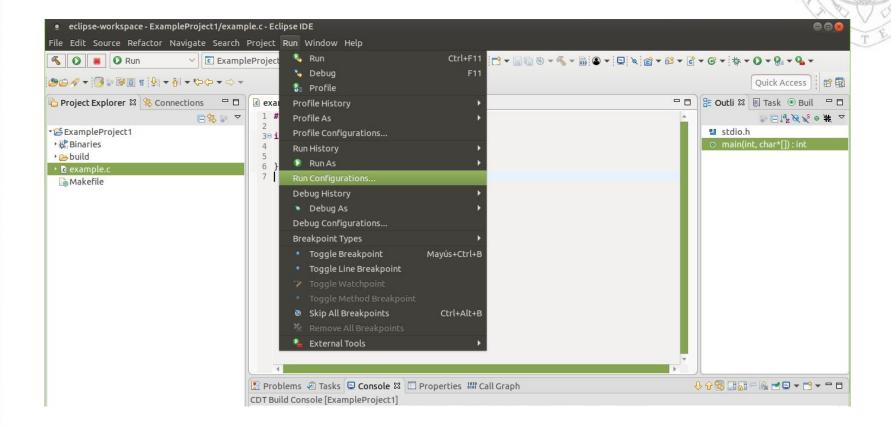


Mensajes de compilación

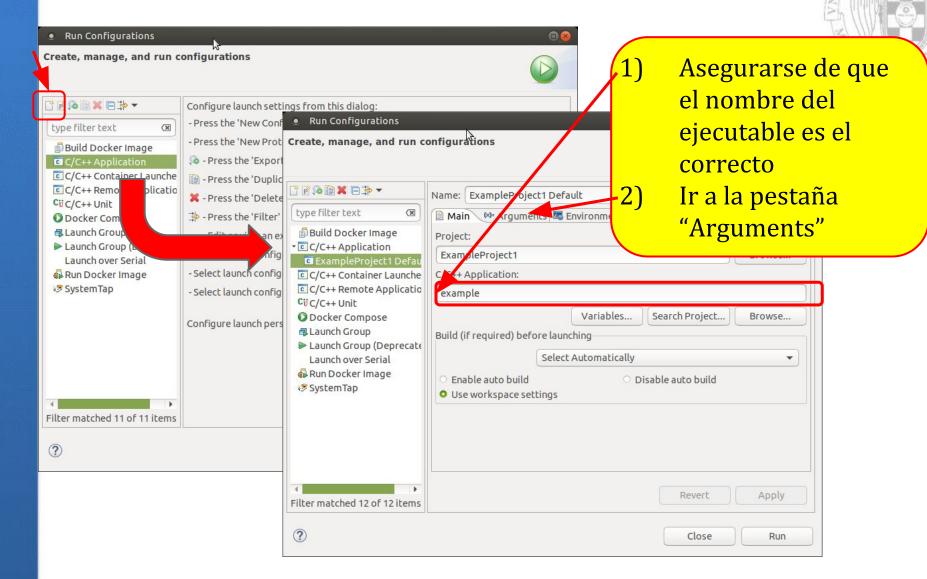




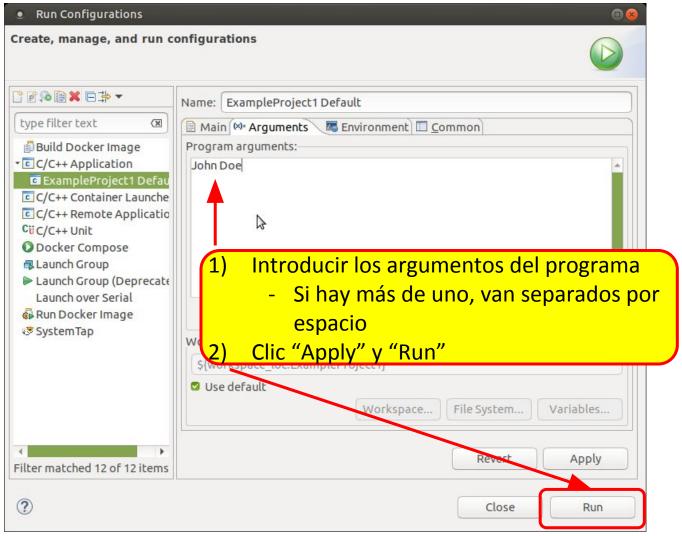
Creando configuración de ejecución



Creando configuración de ejecución



Ejecución del programa

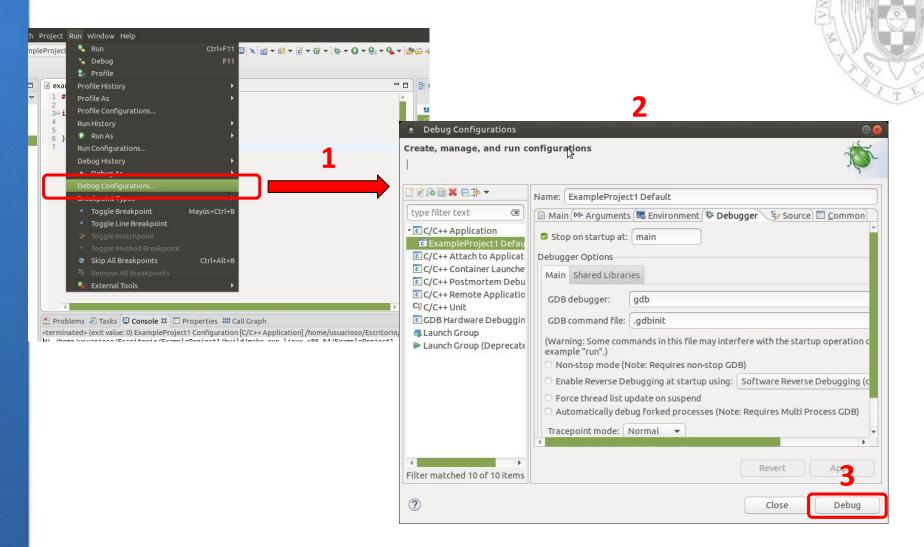




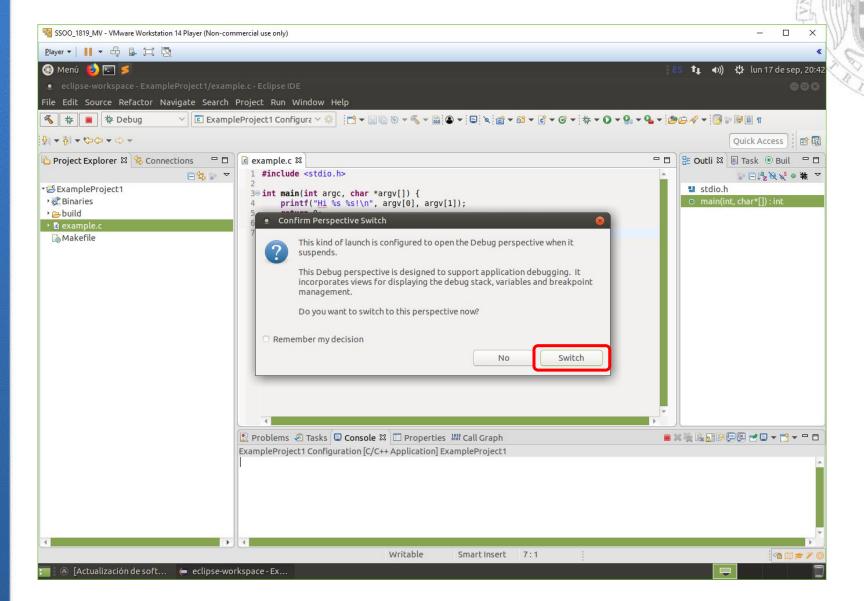
Ejecución del programa



Creando configuración de depuración

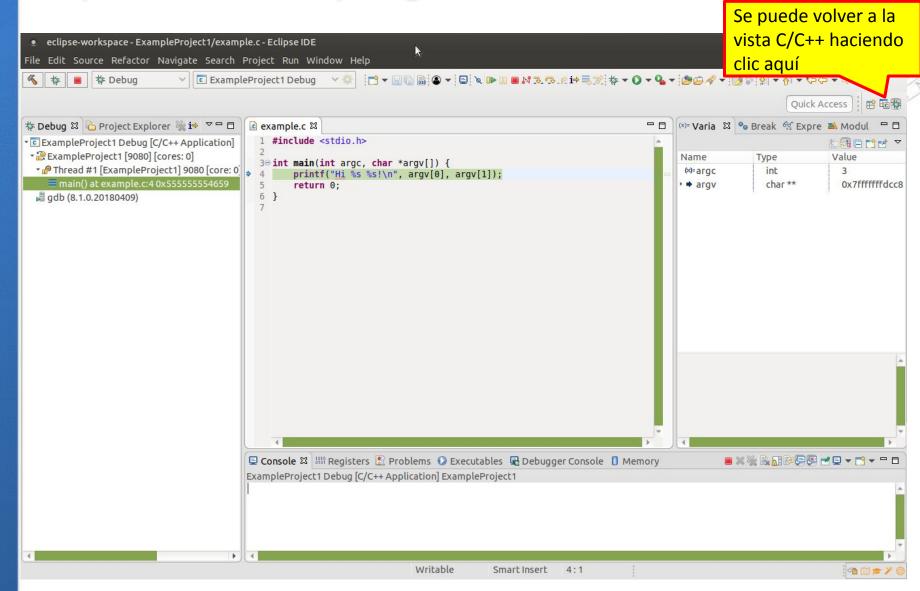


Creando configuración de depuración

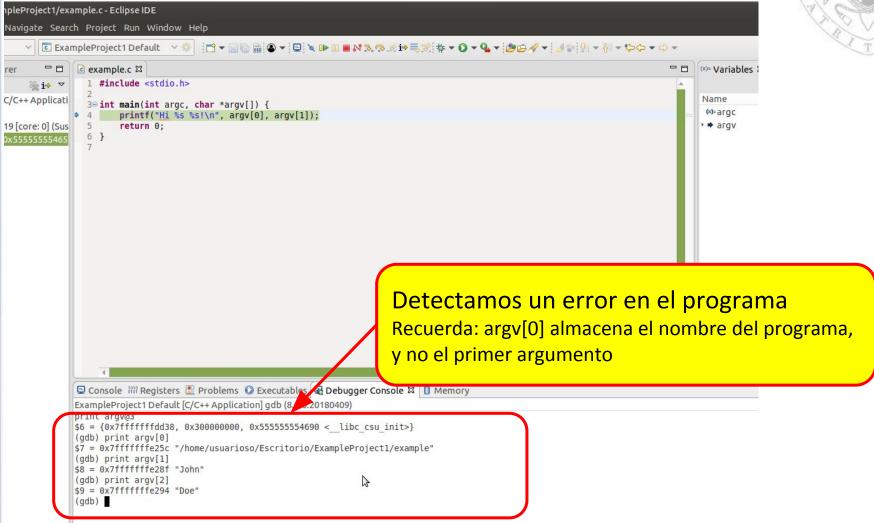




Depuración del programa



Depuración del programa: GDB



Uso de GNU make desde el terminal

