Republic of Tunisia Ministry of Higher Education and Scientific Research

Université de Tunis El Manar

University of Tunis El Manar Higher Institute Of Informatics

ADVANCED OPERATING SYSTEM

PROJECT REPORT

COMPLETED BY :

Nidhal Ghazouani

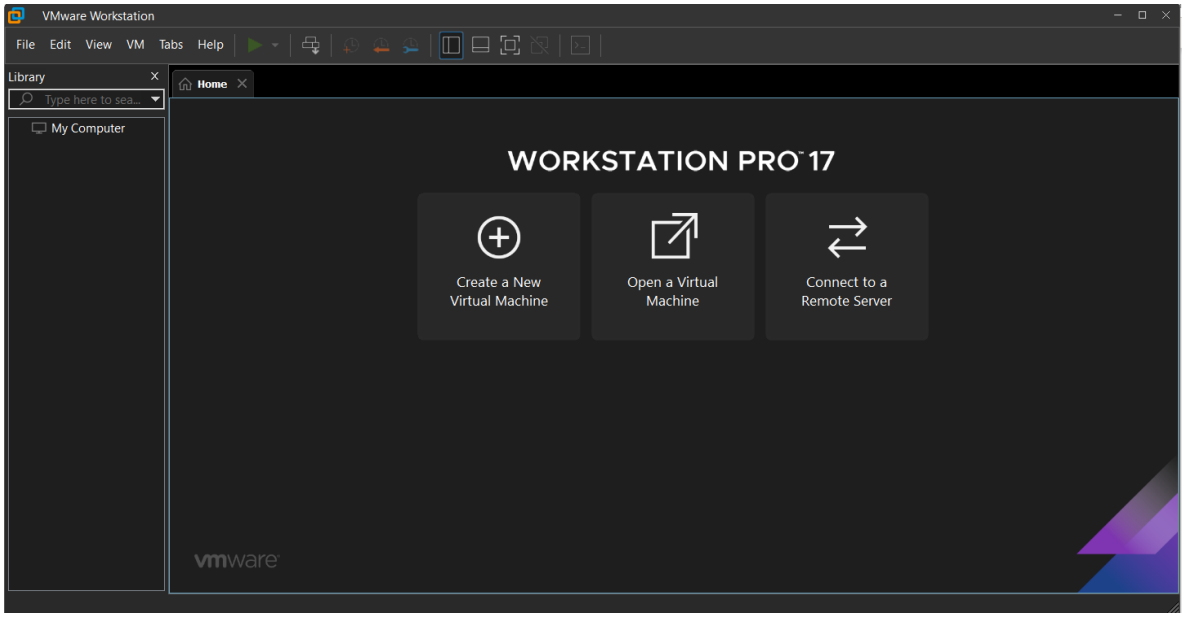
Process Scheduler under Linux

Année Universitaire : 2023-2024

Preparing the travel environment

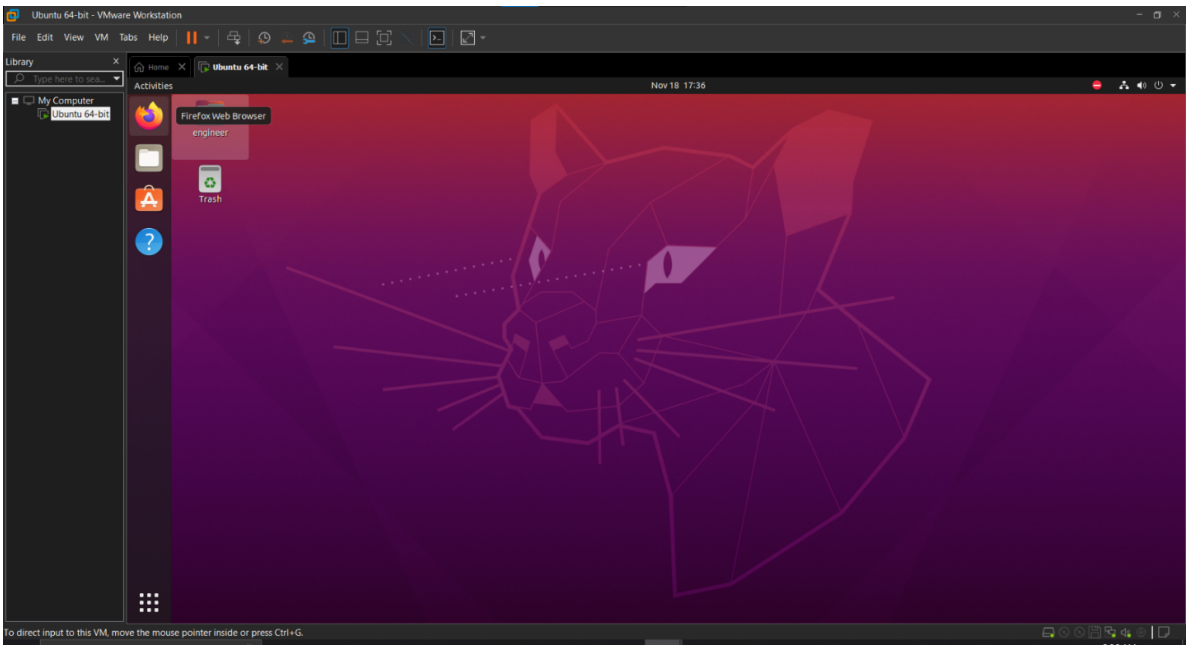
1. Installation of VMware workstation pro 17

As a first step, we installed the VMware workstation software that allows us to create a virtual machine by virtualizing the Ubuntu operating system, and to test our programs in C, in complete security for our already installed system.



2. Installation of virtual machine Ubuntu

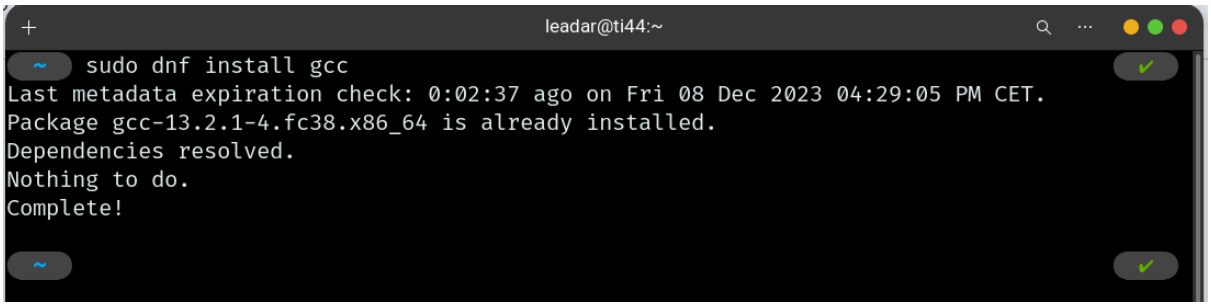
We have set memory size, hard disk, hard disk file type, physical hard disk storage and file location..We added the file ubuntu.iso to install the operating system in our virtual machine. Then we created an Ubuntu virtual machine to commence our work.



3. Installation of gcc (compiler)

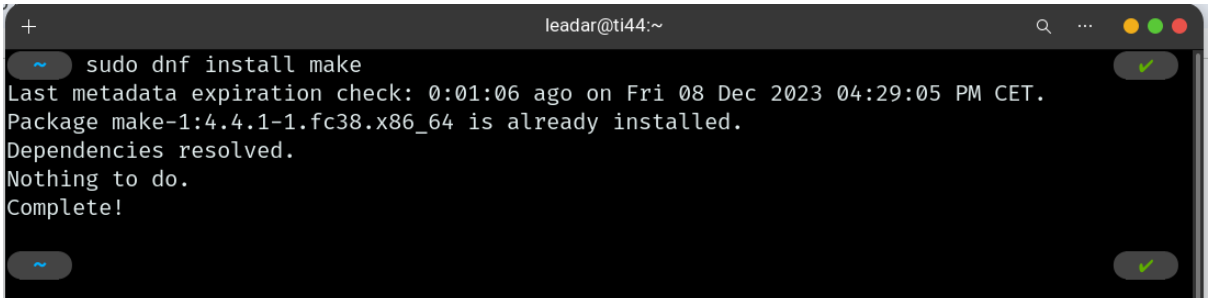
The GCC command in Linux, which stands for GNU Compiler Collection, is an incredibly powerful tool used for compiling and debugging programs written in C, C++, and other languages.

To install gcc we use this command: sudo dnf install gcc.



4. Installation of make

A makefile is a special file, containing shell commands, that we create and name makefile. To install makefile we use this command: sudo apt install make. While in the directory containing this makefile, we type make and the commands in the makefile will be executed as shown in the following capture.



5. Teste compilation of project

Compile the program: Type the command gcc prog.c .a , this command will invoke the GNU C compiler to compile the file prog.c and display (-a) the result in an executable called prog

Run the program: Type the command ./prog it gives the result presented in the following capture.

```
leadar@ti44:~
~ nano prog.c

~ gcc prog.c -o prog

~ ./prog
testing environment gcc

~
```

6. Installing GTK 3

GTK is a widget Toolkit, it's a multi-platform toolkit for creating graphical user interfaces. Offering a complete set of widgets

This is implemented in C using GObject, an object-oriented framework for C.

To install gcc we use this command: sudo dnf install gtk3-devel.

```
leadar@ti44:~
~ sudo dnf install gtk3-devel
Last metadata expiration check: 0:01:11 ago on Thu 07 Dec 2023 07:44:16 PM CET.
Package gtk3-devel-3.24.38-1.fc38.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!

~
```

7. Managing Linux permissions

Managing access to resources is a fundamental task for sysadmins.

This responsibility consists of three components: identities, resources, and permissions.

To set permissions, we use the chmod command in the makefile:

```
all:gui $(SOURCE)  menu
    chmod 755 $(SOURCE) menu
```

755: (rwxr-xr-x) The file's owner may read, write, and execute the file. All others may read and execute the file.

Run The Project

1. Compiling

In the correct directory, open the terminal and type make to execute the makefile, we type make all and the commands in the makefile will be executed as shown in the following capture.
To remove all the executables generated by makefile, we use the command make clean

```
leadar@ti44:~/Documents/projet_se_global/Scheduler_Project_2023/Project
~/Doc/pr/Scheduler_Project_2/Project  make all
gcc `pkg-config --cflags --libs gtk+-3.0` -c Gui/gui_functions.c -o Gui/gui_functions.o
gcc `pkg-config --cflags --libs gtk+-3.0` -Wall -g -w Algorithm/Fifo.c Gui/gui_functions.o
 -o Exe/Fifo
gcc `pkg-config --cflags --libs gtk+-3.0` -Wall -g -w Algorithm/Priority.c Gui/gui_functio
ns.o -o Exe/Priority
gcc `pkg-config --cflags --libs gtk+-3.0` -Wall -g -w Algorithm/RoundRobin.c Gui/gui_funct
ions.o -o Exe/RoundRobin
gcc `pkg-config --cflags --libs gtk+-3.0` -Wall -g -w Algorithm/Srtf.c Gui/gui_functions.o
 -o Exe/Srtf
gcc -Wall -g -w menu.c -o menu
chmod 755 Exe/Fifo Exe/Priority Exe/RoundRobin Exe/Srtf menu

~/Doc/pr/Scheduler_Project_2/Project
```

2. Executing

Then we launch the executable menu we type ./menu you will be directed to the "Application Menu" where you will see all of the existing algorithms.
Next, you must choose a Scheduling Algorithm.

```
./menu processes_list.txt
~/Doc/pr/Scheduler_Project_2/Project  ./menu processes_list.txt

        ------------------------------------------------------
         M i n i m a l   P r o c e s s   S c h e d u l e r   V 1 . 0
        ------------------------------------------------------

Do you want to generate a random process list [Y] , or keep the same process list [N]
```

if we press [Y] , it will ask us to enter “the number of processes” that we want to schedule, secondly will prompt us to enter “the range of Burst time” , “Arrival time” , “priority” (min,max) ….
then based on the input it will randomly generate a list of processes and will save the list in a file that was passed from the argument “processes_list.txt” , in case the file does not exist our application will create a new file then save the content in it

```
./menu processes_list.txt
~/Doc/pr/Scheduler_Project_V/S/Project  ./menu processes_list.txt

                ------------------------------------------------------
                 M i n i m a l   P r o c e s s   S c h e d u l e r   V 1 . 0
                ------------------------------------------------------

Do you want to generate a random process list [Y] , or keep the same process list [N]
Y
Enter the number of process you want to schedule
5
** Required range of process arrival time **
Enter the minimum value of process arrival time
1
Enter the maximum value of process' arrival time
10
** Required the range of process execution time **
Enter the minimum value of process execution time
1
Enter the maximum value of process execution time
5
** Required the range of process priority **
Enter the minimum value of process priority
1
Enter the maximum value of process priority
10
Name : P0 , Arrival time : 5 , Execution time : 4 , Priority : 1 , PID : 0
Name : P1 , Arrival time : 2 , Execution time : 2 , Priority : 2 , PID : 1
Name : P2 , Arrival time : 1 , Execution time : 4 , Priority : 1 , PID : 2
Name : P3 , Arrival time : 5 , Execution time : 1 , Priority : 9 , PID : 3
Name : P4 , Arrival time : 3 , Execution time : 5 , Priority : 3 , PID : 4
```

Note: If We press [N] this will keep the file provided from argument and will read data from it directly , without the need to generate random processes

If you choose Fifo: Fifo (First In First Out), as indicated by his name, it executes first the job that first entered the queue

Result : cli mode

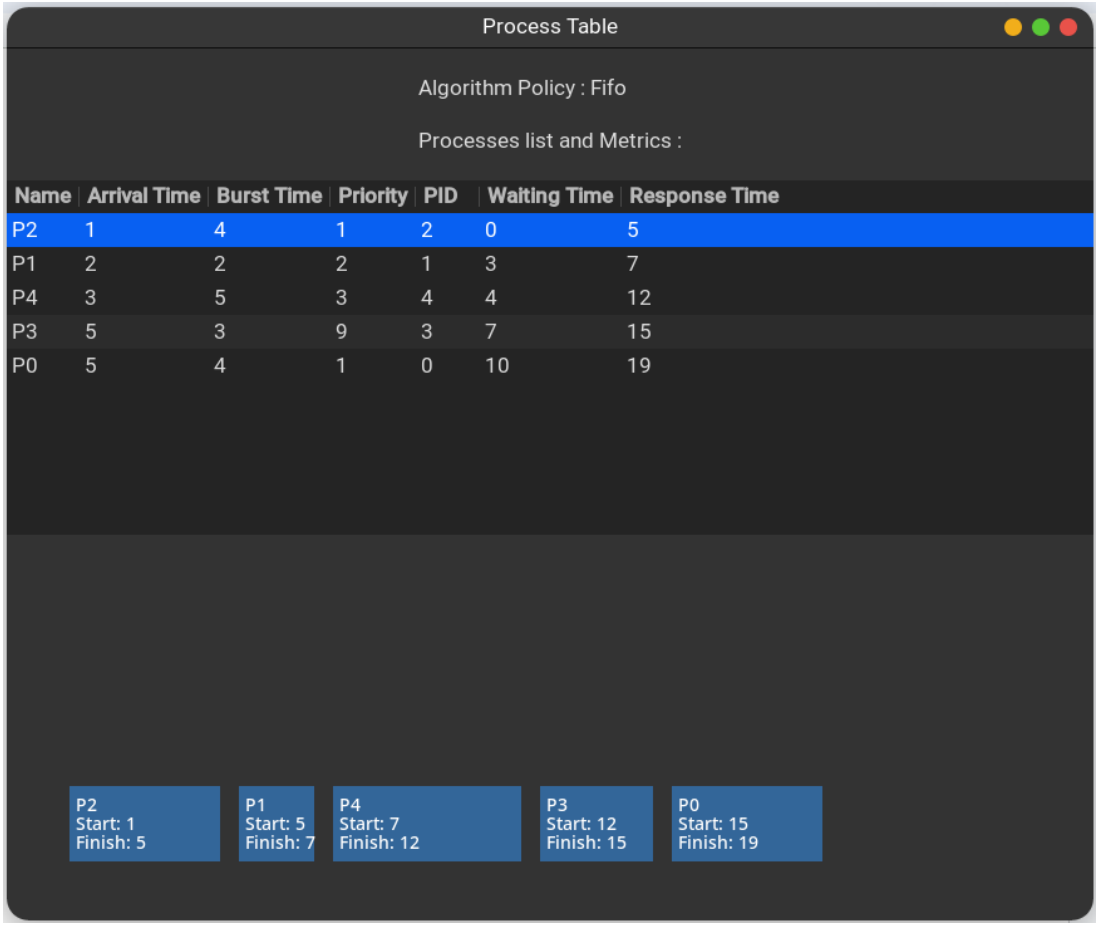
```
./menu processes_list.txt

+-----+-------------+------------+---------------+---------------+
|Name | Arrival Date| Burst Time | Waiting Time  | Response Time |
+-----+-------------+------------+---------------+---------------+
|  P2 |       1     |       4    |        0      |        5      |
+-----+-------------+------------+---------------+---------------+
|  P1 |       2     |       2    |        3      |        7      |
+-----+-------------+------------+---------------+---------------+
|  P4 |       3     |       5    |        4      |       12      |
+-----+-------------+------------+---------------+---------------+
|  P3 |       5     |       3    |        7      |       15      |
+-----+-------------+------------+---------------+---------------+
|  P0 |       5     |       4    |       10      |       19      |
+-----+-------------+------------+---------------+---------------+

 GANTT Diagram

 -------- ---- ---------- ------ --------
|   P2   | P1 |    P4    |  P3  |   P0   |
 -------- ---- ---------- ------ --------
1        5    7         12     15       19
```

Result : Graphic mode



Name	Arrival Time	Burst Time	Priority	PID	Waiting Time	Response Time
P2	1	4	1	2	0	5
P1	2	2	2	1	3	7
P4	3	5	3	4	4	12
P3	5	3	9	3	7	15
P0	5	4	1	0	10	19

END.