



Sistemas operativos

Algoritmo Prioridades

Alumno:

1.- Omar Santos Bernabé  
[osb0005@alu.ubu.es](mailto:osb0005@alu.ubu.es)

Profesores:

José Manuel Sáiz

Enrique Sierra

Leticia Curiel

Alumnos 2014-2015:

ANTON GARCIA, ADRIÁN

BAILADOR PANERO, ADRIÁN

BASURTO, NUÑO

BENITO, JUAN FRANCISCO

BERLINCHES, ERIC

CASTRO HURTADO, VÍCTOR

GARCÍA GUTIÉRREZ, ÁLVARO

GORDO ORIVE, ZAIRA

MERINO, SHEILA

MILLÁN, MARCOS

SAIZ, FRANCISCO

ALEJANDRO CABALLERO GIL

RUBÉN URUÑUELA HERRERA

ADRIÁN MARCOS BATLLE,

IVÁN IGLESIAS CUESTA

ÁLVARO RUIFERNANDEZ PALACIOS

Alumnos 2015-2016:

OMAR SANTOS BERNABÉ

# Índice de contenido

1. Algoritmo con el que se trabaja y descripción:.....	4
2. Ejercicios de ejemplo:.....	5
2.1. Ejercicio hecho a mano:.....	5
Ejercicio1-Menor.....	5
Ejercicio2-Menor.....	6
Ejercicio1-Mayor.....	7
Ejercicio2-Menor.....	8
2.2.Ejercicio hecho con el algoritmo.....	9
Ejercicio1-Menor.....	9
Ejercicio2-Menor.....	14
Ejercicio1-Mayor.....	19
Ejercicio2-Mayor.....	24
3. Datos de interés sobre el código.....	29

## **1. Algoritmo con el que se trabaja y descripción:**

### **Prioridades**

Este algoritmo basa su planificación en asignar distintas importancias a los procesos, es decir, asignar a cada proceso una determinada prioridad. Este planificador de procesos se implementará tanto apropiativo como no apropiativo. En nuestro caso será no apropiativo.

El orden de ejecución de los procesos seguirá una planificación FIFO en función de la ordenación de ejecución de los procesos por su prioridad.

## 2. Ejercicios de ejemplo:

### 2.1. Ejercicio hecho a mano:

#### Ejercicio1-Menor

Realizando el algoritmo de forma manual, con los siguientes datos, que son los mismos que introduciremos en la ejecución del algoritmo a ordenador, nos quedaría tal que así:

Procesos	Prioridad	Ejecución
P1	1	3
P2	2	7
P3	1	8
P4	2	4

El resultado final de la ejecución sería el siguiente:

P3	P3	P3	P3	P3	P3	P3	P3	P3	P1	P1	P1	P2	P2	P2	P2	P2	P2	P2	P4	P4	P4	P4
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

El orden ejecución de los procesos será en función de la prioridad asignada a los procesos. En este caso al seguir una prioridad menor el primer proceso en ejecutarse será el proceso P3 durante 8 ráfagas de CPU y se ejecuta antes que el proceso P1 porque aunque su prioridad es la misma entra en ejecución primero este proceso porque tiene más tiempo de ejecución. El siguiente proceso en ejecutarse será el proceso P1. Este proceso se encontrara en CPU durante 3 ráfagas. Ahora se encontrara en CPU el proceso P2 y no el proceso P4 ya que en caso de misma prioridad tomara el control el proceso con mayor tiempo de ejecución como pasaba con P1 y P3. El proceso P2 se encontrara en CPU durante 7 ráfagas. Finalmente el proceso P4 tomara el control de CPU durante su ráfaga, en este caso durante cuatro unidades de tiempo.

## Ejercicio2-Menor

Realizando el algoritmo de forma manual, con los siguientes datos, que son los mismos que introduciremos en la ejecución del algoritmo a ordenador, nos quedaría tal que así:

Procesos	Prioridad	Ejecución
P1	3	10
P2	1	1
P3	3	2
P4	4	1
P5	2	5
P6	7	2
P7	12	1
P8	1	4
P9	1	2
P10	2	2

El resultado final de la ejecución sería el siguiente:

P 8	P 8	P 8	P 8	P 9	P 9	P 2	P 5	P 5	P 5	P 5	P 5	P 1 0	P 1 0	P 1	P 1	P 1	P 1	P 1	P 1	P 1	P 1	P 1	P 1	P 3	P 3	P 4	P 6	P 6	P 7
--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	-------------	-------------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------

El orden ejecución de los procesos será en función de la prioridad asignada a los procesos. En este caso al seguir una prioridad menor el primer proceso en ejecutarse será el proceso P8 durante 4 ráfagas de CPU. Se ejecuta antes que el proceso P9 y el proceso P2 porque en el caso de tener misma prioridad toma el control el proceso que mayor tiempo de ejecución tenga. El siguiente proceso en ejecutarse será el proceso P9 durante dos ráfagas de CPU y una vez termine se ejecutará el proceso P2 durante una ráfaga. El siguiente en ejecutarse es el proceso P5 puesto que es el que mayor tiempo de ejecución tiene de los de prioridad dos y ejecutándose durante cinco ráfagas de CPU. A continuación se ejecuta el proceso P10 durante dos ráfagas de ejecución. El siguiente proceso en ejecutarse es el proceso P1 que ya tiene prioridad tres y tiene mayor tiempo de ejecución que el proceso P3 y esta ejecutándose durante 10 ráfagas de tiempo. A continuación se ejecuta el proceso P3 durante dos ráfagas de CPU. Luego se ejecuta el proceso P4 con prioridad cuatro durante una ráfaga de tiempo. A continuación el proceso P6 durante dos unidades de tiempo. Y finalmente se ejecuta el proceso P7 por tener la mayor prioridad de todos.

## Ejercicio1-Mayor

Realizando el algoritmo de forma manual, con los siguientes datos, que son los mismos que introduciremos en la ejecución del algoritmo a ordenador, nos quedaría tal que así:

Procesos	Prioridad	Ejecución
P1	1	3
P2	2	7
P3	1	8
P4	2	4

El resultado final de la ejecución sería el siguiente:

P2	P2	P2	P2	P2	P2	P2	P4	P4	P4	P4	P3	P3	P3	P3	P3	P3	P3	P3	P1	P1	P1
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

El orden ejecución de los procesos será en función de la prioridad asignada a los procesos. En este caso al seguir una prioridad mayor el primer proceso en ejecutarse será el proceso P2 durante siete ráfagas de CPU y se ejecuta antes que el proceso P4 porque aunque su prioridad es la misma entra en ejecución primero este proceso porque tiene más tiempo de ejecución. El siguiente proceso en ejecutarse será el proceso P4. Este proceso se encontrara en CPU durante cuatro ráfagas. Ahora se encontrara en CPU el proceso P3 y no el proceso P1 ya que en caso de misma prioridad tomara el control el proceso con mayor tiempo de ejecución como pasaba con P2 y P4. El proceso P3 se encontrara en CPU durante ocho ráfagas. Finalmente el proceso P1 tomara el control de CPU durante su ráfaga, en este caso durante tres unidades de tiempo.

## Ejercicio2-Menor

Realizando el algoritmo de forma manual, con los siguientes datos, que son los mismos que introduciremos en la ejecución del algoritmo a ordenador, nos quedaría tal que así:

Procesos	Prioridad	Ejecución
P1	3	10
P2	1	1
P3	3	2
P4	4	1
P5	2	5
P6	7	2
P7	12	1
P8	1	4
P9	1	2
P10	2	2

El resultado final de la ejecución sería el siguiente:

P 7	P 6	P 6	P 4	P 1	P 1	P 1	P 1	P 1	P 1	P 1	P 1	P 1	P 1	P 3	P 3	P 5	P 5	P 5	P 5	P 5	P 1 0	P 1 0	P 8	P 8	P 8	P 8	P 9	P 9	P 2
--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	-------------	-------------	--------	--------	--------	--------	--------	--------	--------

El orden ejecución de los procesos será en función de la prioridad asignada a los procesos. En este caso al seguir una prioridad mayor el primer proceso en ejecutarse será el proceso P7 durante una ráfaga de CPU. Se ejecuta a continuación el proceso P6 por ser el siguiente con mayor prioridad, y según termina sus dos unidades de tiempo de ejecución se ejecuta el proceso P4 durante una ráfaga. A continuación se ejecuta el proceso P1 que de los que tienen mayor prioridad es el que mayor tiempo de ejecución tiene, y según termina se ejecuta el proceso P3 que tiene su misma prioridad pero menor tiempo de ejecución. A continuación se ejecuta el proceso P5 durante cinco ráfagas de tiempo porque tiene la misma prioridad que el proceso P10 pero mayor tiempo de ejecución. El siguiente proceso que se ejecuta es el proceso P10 durante dos unidades de tiempo. Por último se ejecutan los que menor prioridad tiene, y de estos el primero en ejecutarse es el proceso P8 porque tiene un mayor tiempo de ejecución que los otros dos procesos. El siguiente es el proceso P9 durante dos ráfagas de CPU. Y final mente de ejecuta el proceso P2 durante una ráfaga.



## 2.2.Ejercicio hecho con el algoritmo

### Ejercicio1-Menor

A continuación veremos mediante volcados de pantalla la ejecución del algoritmo con los datos del primer ejercicio realizado a mano:

Lo primero de todo que se tiene que realizar es ir a la carpeta o directorio en el que se encuentre el algoritmo y una vez estemos ahí situados utilizaremos el comando “./PlanificacionPRIORIDADES\_MENOR.sh” para ejecutarlo. Una vez hemos ejecutado el algoritmo lo primero que nos aparecerá sera esto.

Aparece una licencia de creative commons que indica que es una versión nuestra y que no puede ser modificada sin autorización y que no puede usarse con un fin comercial. A continuación nos aparece la cabecera del algoritmo en la que podemos encontrar el nombre del algoritmo, el nombre del alumno que lo ha modificado por última vez y la versión a la que corresponde el algoritmo, en este caso la última versión realizada que es la de Junio del 2015.

Como se podía ver en la imagen anterior al final nos pedía introducir el número de procesos que queremos ejecutar. Introducimos un total de cuatro procesos para que coincida con el número de procesos que hemos seleccionado en el primer ejercicio que hemos hecho de forma manual. A continuación nos pregunta si queremos introducir los datos de forma manual. En el caso de introducir como respuesta una s nos pedirá como se vera a continuación paso a paso cada uno de los datos necesarios

para cada uno de los procesos. Si por el contrario introducimos una n, como se verá en el próximo ejemplo, cogerá los datos del fichero datosEntradaPredeterminados.txt en el que hay datos para varios procesos.

Proceso	T. Ejecución	Prioridad
P1	3	1
P2	7	2
P3	8	1
P4	4	

Prioridad del proceso 3  
2

Lo primero que introducíamos como se ve en la imagen anterior era el nombre para el primer proceso y una vez introducido nos aparecía la siguiente tabla en la que podemos ver de forma mas clara los datos que ya han sido introducidos para así evatar introducir los mismos tiempos de llegada y de ejecución para distintos procesos, lo que sería inútil introducir varios procesos con los mismos datos puesto que de ese modo no conseguiríamos ver bien el funcionamiento de este algoritmo.

Como se puede observar se introducen los datos de forma ordenada cada uno para su procesos, y el algoritmo nos va pidiendo que introduzcamos primero el nombre que queremos para el proceso, a continuación el tiempo de llegada, el instante en el que dicho proceso llega para ejecutarse y finalmente el tiempo de ejecución que indica el tiempo que estará ejecutandose dicho proceso. Se realiza esto para cada uno de los procesos, en este caso para cuatro procesos que es el número de procesos que queríamos ejecutar y una vez hemos introducido los datos paraa todos los procesos el algoritmo se dispone a ejecutarse.

```
Proceso: P1 Tiempo de ejecución: 3 Prioridad del Proceso: 1
Proceso: P2 Tiempo de ejecución: 7 Prioridad del Proceso: 2
Proceso: P3 Tiempo de ejecución: 8 Prioridad del Proceso: 1
Proceso: P4 Tiempo de ejecución: 4 Prioridad del Proceso: 2
```

Lo primero que aparece una vez hemos terminado de introducir los datos de todos los procesos es una pequeña lista en la que podemos encontrarnos el nombre del proceso, el tiempo de llegada y el tiempo de ejecución de cada uno de los procesos que hemos introducido antes. De seguido podemos ver el orden de ejecución de los diferentes procesos en una pequeña tabla que contiene los procesos, su tiempo de llegada, su tiempo de ejecución, su tiempo de espera y su tiempo de respuesta.

Proceso	Ejecución	Prioridad	Espera	Respuesta
* P3 *	* 8 *	* 1 *	* 0 *	* 8 *
* P1 *	* 3 *	* 1 *	* 8 *	* 11 *
* P2 *	* 7 *	* 2 *	* 11 *	* 18 *
* P4 *	* 4 *	* 2 *	* 18 *	* 22 *
* T.espera medio: 9 ~ * T.retorno medio: 14				

Como podemos observar el primer proceso en ejecutarse es el proceso P3 que tiene prioridad uno y como el otro proceso con prioridad uno P1 tiene menor tiempo de ejecución comienza P3 a ejecutarse, por lo que tiene cero unidades de tiempo de espera y como su ejecución es de ocho unidades de tiempo pues su tiempo de respuesta es de ocho unidades. En el momento que termina el proceso P3 comienza la ejecución. Su tiempo de ejecución era de tres unidades de tiempo y tenía una espera de ocho unidades que se debía a que el proceso P3 se había ejecutado antes que este, por lo que su tiempo de respuesta final era de once unidades de tiempo ya que se suman su ejecución y su tiempo de espera. El siguiente proceso que entra a ejecutarse una vez termina de ejecutarse el proceso P1 es el proceso P2, que tiene prioridad dos y comienza a ejecutarse antes del proceso P4 que también tiene prioridad dos porque su tiempo de ejecución es mayor que el de P4. Por ese motivo comienza la ejecución de P2 que tiene un tiempo de ejecución de siete unidades de tiempo y que tiene un tiempo de espera de once unidades, por lo que su tiempo final de respuesta es de dieciocho unidades de tiempo que sale de sumar su espera con su tiempo de ejecución. Finalmente se ejecuta el proceso P4 que tenía un tiempo de ejecución de cuatro unidades de tiempo y una espera de dieciocho por lo que su tiempo de respuesta final era de veintidós unidades de tiempo. Con el resultado de realizar el algoritmo podemos conocer los tiempos de espera medios que en este caso es de nueve unidades de tiempo y el tiempo de retorno medio que es de catorce unidades para este ejemplo.

¿Quieres abrir el informe? ([s],n): s

Por último se nos da la opción de abrir o no un informe con el resultado de la ejecución del algoritmo. A la hora de abrir el informe nos da dos opciones, introducir una s para abrirlo, que de igual forma funciona introduciendo una S o simplemente pulsando intro con el hueco vacío, o una n o N para no abrirlo y terminar la ejecución del algoritmo. En el caso de abrir el informe se nos abrirá un archivo txt como el siguiente.

```

informePrioridadMenor.txt
#####
#                               Creative Commons                               #
#                               #                                              #
#                               BY - Atribución (BY)                            #
#                               NC - No uso Comercial (NC)                     #
#                               SA - Compartir Igual (SA)                      #
#                               #                                              #
#####
#                               #                                              #
#                               INFORME DE PRÁCTICA                           #
#                               GESTIÓN DE PROCESOS                           #
#                               -----                                       #
#                               Nuevos alumnos:                               #
#                               Alumnos: Omar Santos Bernabé                  #
#                               Sistemas Operativos 2º Semestre                #
#                               Grado en ingeniería informática (2015-2016)    #
#                               #                                              #
#####

Proceso: P1 Tiempo de ejecución: 3 Prioridad de Proceso: 1
Proceso: P2 Tiempo de ejecución: 7 Prioridad de Proceso: 2
Proceso: P3 Tiempo de ejecución: 8 Prioridad de Proceso: 1
Proceso: P4 Tiempo de ejecución: 4 Prioridad de Proceso: 2

```

En el informe se puede ver de nuevo al principio la licencia del creative commons que como he explicado antes nos indica que el algoritmo no es de uso comercial y que no se debe modificar, sino que debe entregarse como se a recibido.

A continuación se observa una cabecera que indica el nombre del último alumno que ha realizado esta versión del algoritmo y el año o curso en el cual se ha realizado dicha versión.

Podemos observar los datos que se han introducido para la realización de este ejercicio en el orden que han sido introducidos. Los datos de muestran de la siguiente forma:

Proceso P1<sup>1</sup> Tiempo de ejecución 3<sup>2</sup> Prioridad del proceso 1<sup>3</sup>

<sup>1</sup>.-Es el nombre que hemos dado al proceso.

<sup>2</sup>.-Es el tiempo que tarde en ejecutarse dicho proceso.

<sup>3</sup>.- Es la prioridad que tiene dicho proceso.

	Proceso		Ejecución		Prioridad		Espera		Respuesta	
*	P3	*	8	*	1	*	0	*	8	*
*	P1	*	3	*	1	*	8	*	11	*
*	P2	*	7	*	2	*	11	*	18	*
*	P4	*	4	*	2	*	18	*	22	*

\* T.espera medio: 9 - \* T.retorno medio: 14

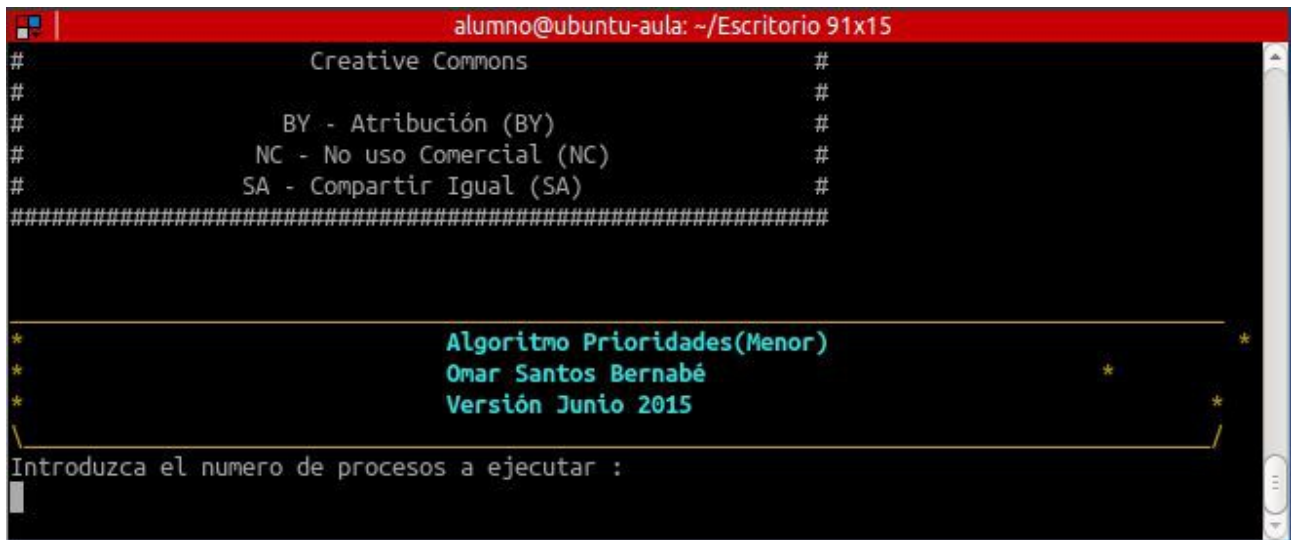
Finalmente observamos los resultados de la ejecución del algoritmo en la cual aparecen los procesos ordenados según su salida de ejecución junto con los datos del tiempo de ejecución y su prioridad, y con el tiempo de espera y el tiempo de respuesta de cada uno de los procesos. Al final aparecen los tiempos de espera medio y de retorno medio del algoritmo a partir de los datos que hemos introducido.

Una vez cerramos el fichero de texto se termina la ejecución del algoritmo.

## Ejercicio2-Menor

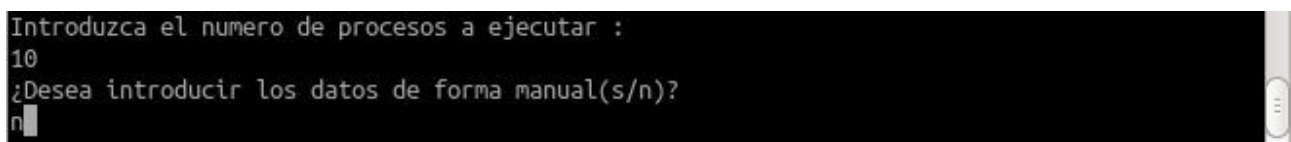
A continuación veremos mediante volcados de pantalla la ejecución del algoritmo con los datos del primer ejercicio realizado a mano:

Lo primero de todo que se tiene que realizar es ir a la carpeta o directorio en el que se encuentre el algoritmo y una vez estemos ahí situados utilizaremos el comando “./PlanificacionPRIORIDADES\_MENOR.sh” para ejecutarlo. Una vez hemos ejecutado el algoritmo lo primero que nos aparecerá será esto.



```
alumno@ubuntu-aula: ~/Escritorio 91x15
# Creative Commons #
# #
# BY - Atribución (BY) #
# NC - No uso Comercial (NC) #
# SA - Compartir Igual (SA) #
#####
* Algoritmo Prioridades(Menor) *
* Omar Santos Bernabé *
* Versión Junio 2015 *
Introduzca el numero de procesos a ejecutar :
```

Aparece una licencia de creative commons que indica que es una versión nuestra y que no puede ser modificada sin autorización y que no puede usarse con un fin comercial. A continuación nos aparece la cabecera del algoritmo en la que podemos encontrar el nombre del algoritmo, el nombre del alumno que lo ha modificado por última vez y la versión a la que corresponde el algoritmo, en este caso la última versión realizada que es la de Junio del 2015.



```
Introduzca el numero de procesos a ejecutar :
10
¿Desea introducir los datos de forma manual(s/n)?
n
```

Como se podía ver en la imagen anterior al final nos pedía introducir el número de procesos que queremos ejecutar. Introducimos un total de 10 procesos para que coincida con el número de procesos que hemos seleccionado en el segundo ejercicio que hemos hecho de forma manual. A continuación nos pregunta si queremos introducir los datos de forma manual. Introducimos una n para indicar que queremos obtener los datos del fichero de datosEntradaPredeterminados.txt en el que hay datos para un alto número de procesos.



```
Introduzca el nombre del proceso 0:
Tiempo De ejecucion del proceso 0:
Prioridad del proceso 0:
Introduzca el nombre del proceso 1:
Tiempo De ejecucion del proceso 1:
Prioridad del proceso 1:
Introduzca el nombre del proceso 2:
Tiempo De ejecucion del proceso 2:
Prioridad del proceso 2:
Introduzca el nombre del proceso 3:
Tiempo De ejecucion del proceso 3:
Prioridad del proceso 3:
Introduzca el nombre del proceso 4:
Tiempo De ejecucion del proceso 4:
Prioridad del proceso 4:
Introduzca el nombre del proceso 5:
Tiempo De ejecucion del proceso 5:
Prioridad del proceso 5:
Introduzca el nombre del proceso 6:
Tiempo De ejecucion del proceso 6:
Prioridad del proceso 6:
Introduzca el nombre del proceso 7:
Tiempo De ejecucion del proceso 7:
Prioridad del proceso 7:
Introduzca el nombre del proceso 8:
Tiempo De ejecucion del proceso 8:
Prioridad del proceso 8:
Introduzca el nombre del proceso 9:
Tiempo De ejecucion del proceso 9:
Prioridad del proceso 9:
```

Como se puede observar en la imagen el propio algoritmo va cogiendo en orden los datos que necesita para los diez procesos que vamos a ejecutar. Los datos se recogen en el mismo orden que se pedían de forma manual, primero el nombre que queremos dar al proceso, después el tiempo de llegada de dicho proceso y finalmente el tiempo de ejecución para ese proceso. Se realiza ese paso hasta recoger los datos para los diez procesos que queremos.

Se introducen los datos de forma ordenada cada uno para su procesos, y el algoritmo nos va pidiendo que introduzcamos primero el nombre que queremos para el proceso, a continuación el tiempo de ejecución, que indica el tiempo que estará ejecutándose dicho proceso y finalmente su prioridad. Se realiza esto para cada uno de los procesos, en este caso para diez procesos que es el número de procesos que queríamos ejecutar y una vez ha recogido los datos del fichero para todos los procesos el algoritmo se dispone a ejecutarse.

```

Proceso: P1 Tiempo de ejecución: 10 Prioridad de Proceso: 3
Proceso: P2 Tiempo de ejecución: 1 Prioridad de Proceso: 1
Proceso: P3 Tiempo de ejecución: 2 Prioridad de Proceso: 3
Proceso: P4 Tiempo de ejecución: 1 Prioridad de Proceso: 4
Proceso: P5 Tiempo de ejecución: 5 Prioridad de Proceso: 2
Proceso: P6 Tiempo de ejecución: 2 Prioridad de Proceso: 7
Proceso: P7 Tiempo de ejecución: 1 Prioridad de Proceso: 12
Proceso: P8 Tiempo de ejecución: 4 Prioridad de Proceso: 1
Proceso: P9 Tiempo de ejecución: 2 Prioridad de Proceso: 1
Proceso: P10 Tiempo de ejecución: 2 Prioridad de Proceso: 2

```

Lo primero que aparece una vez hemos terminado de introducir los datos de todos los procesos es una pequeña lista en la que podemos encontrarnos el nombre del proceso, el tiempo de ejecución y la prioridad de cada uno de los procesos que hemos introducido antes. De seguido podemos ver el orden de ejecución de los diferentes procesos en una pequeña tabla que contiene los procesos, su tiempo de llegada, su tiempo de ejecución, su tiempo de espera y su tiempo de respuesta.

Proceso	Ejecución	Prioridad	Espera	Respuesta
* P8 *	4	1	0	4
* P9 *	2	1	4	6
* P2 *	1	1	6	7
* P5 *	5	2	7	12
* P10 *	2	2	12	14
* P1 *	10	3	14	24
* P3 *	2	3	24	26
* P4 *	1	4	26	27
* P6 *	2	7	27	29
* P7 *	1	12	29	30
* T.espera medio: 14 - * T.retorno medio: 17				

El orden ejecución de los procesos será en función de la prioridad asignada a los procesos. En este caso al seguir una prioridad menor el primer proceso en ejecutarse será el proceso P8 durante 4 ráfagas de CPU. Se ejecuta antes que el proceso P9 y el proceso P2 porque en el caso de tener misma prioridad toma el control el proceso que mayor tiempo de ejecución tenga. Como ningún proceso se ejecuta antes que este su tiempo de espera es de cero unidades de tiempo, por lo que su tiempo de respuesta será igual que su tiempo de ejecución, en este caso de cuatro unidades de tiempo. El siguiente proceso en ejecutarse será el proceso P9 durante dos ráfagas de CPU y como tiene un tiempo de espera de cuatro pues su tiempo de ejecución final es de seis unidades. Una vez termine se ejecutará el proceso P2 durante una ráfaga ya que es el último de los que tienen prioridad uno y tiene un tiempo de espera de seis unidades y por consecuente un tiempo de respuesta total de siete, puesto que este tiempo es la suma del tiempo de espera(seis) y el tiempo de ejecución(unos). El siguiente en ejecutarse es el proceso P5 puesto que es el que mayor tiempo de ejecución tiene de los procesos de prioridad dos y se ejecuta durante cinco ráfagas de CPU con un tiempo final de respuesta de doce unidades de tiempo puesto que su tiempo de espera era de siete unidades. A continuación se ejecuta el proceso P10 durante dos ráfagas de



ejecución y como su tiempo de espera era de doce unidades de tiempo su tiempo de respuesta suma a catorce, resultado de la suma de su tiempo de ejecución y su tiempo de espera. El siguiente proceso en ejecutarse es el proceso P1 que ya tiene prioridad tres y tiene mayor tiempo de ejecución que el proceso P3 y esta ejecutándose durante 10 ráfagas de tiempo con un tiempo de espera de catorce unidades y un tiempo total de respuesta de veinticuatro unidades de tiempo. A continuación se ejecuta el proceso P3 durante dos ráfagas de CPU con un tiempo de espera de veinticuatro unidades de tiempo y por consecuente un tiempo de respuesta final de veintiseis unidades. Luego se ejecuta el proceso P4 con prioridad cuatro durante una ráfaga de tiempo. A continuación el proceso P6 durante dos unidades de tiempo y prioridad siete que se ejecuta dejando un tiempo de respuesta final de veintinueve unidades de tiempo. Y finalmente se ejecuta el proceso P7 por tener la mayor prioridad de todos que tiene un tiempo de espera de veintinueve unidades de tiempo y deja un tiempo de respuesta final de treinta unidades porque tiene una unidad de tiempo de ejecución.

```
¿Quieres abrir el informe? ([s],n): s
```

Por último se nos da la opción de abrir o no un informe con el resultado de la ejecución del algoritmo. A la hora de abrir el informe nos da dos opciones, introducir una s para abrirlo, que de igual forma funciona introduciendo una S o simplemente pulsando intro con el hueco vacío, o una n o N para no abrirlo y terminar la ejecución del algoritmo. En el caso de abrir el informe se nos abrirá un archivo txt como el siguiente

```
#####
#                               #
#           Creative Commons    #
#                               #
#           BY - Atribución (BY) #
#           NC - No uso Comercial (NC) #
#           SA - Compartir Igual (SA) #
#####

#####
#                               #
#           INFORME DE PRÁCTICA    #
#           GESTIÓN DE PROCESOS    #
#           -----                #
#           Nuevos alumnos:        #
#           Alumnos: Omar Santos Bernabé #
#           Sistemas Operativos 2º Semestre #
#           Grado en ingeniería informática (2015-2016) #
#                               #
#####
```

Proceso: P1 Tiempo de ejecución: 10 Prioridad de Proceso: 3

Proceso: P2 Tiempo de ejecución: 1 Prioridad de Proceso: 1

Proceso: P3 Tiempo de ejecución: 2 Prioridad de Proceso: 3

Proceso: P4 Tiempo de ejecución: 1 Prioridad de Proceso: 4

Proceso: P5 Tiempo de ejecución: 5 Prioridad de Proceso: 2

Proceso: P6 Tiempo de ejecución: 2 Prioridad de Proceso: 7

Proceso: P7 Tiempo de ejecución: 1 Prioridad de Proceso: 12

Proceso: P8 Tiempo de ejecución: 4 Prioridad de Proceso: 1

Proceso: P9 Tiempo de ejecución: 2 Prioridad de Proceso: 1

Proceso: P10 Tiempo de ejecución: 2 Prioridad de Proceso: 2

	Proceso		Ejecución		Prioridad		Espera		Respuesta	
*	P8	*	4	*	1	*	0	*	4	*
*	P9	*	2	*	1	*	4	*	6	*
*	P2	*	1	*	1	*	6	*	7	*
*	P5	*	5	*	2	*	7	*	12	*
*	P10	*	2	*	2	*	12	*	14	*
*	P1	*	10	*	3	*	14	*	24	*
*	P3	*	2	*	3	*	24	*	26	*
*	P4	*	1	*	4	*	26	*	27	*
*	P6	*	2	*	7	*	27	*	29	*
*	P7	*	1	*	12	*	29	*	30	*

\* T.espera medio: 14 - \* T.retorno medio: 17

En el informe se puede ver de nuevo al principio la licencia del creative commons que como he explicado antes nos indica que el algoritmo no es de uso comercial y que no se debe modificar, sino que debe entregarse como se a recibido.

A continuación se observa una cabecera que indica el nombre del último alumno que ha realizado esta versión del algoritmo y el año o curso en el cual se ha realizado dicha versión.

Podemos observar los datos que se han introducido para la realización de este ejercicio en el orden que han sido introducidos. Los datos de muestran de la siguiente forma:

Proceso P1<sup>1</sup> Tiempo de ejecución 3<sup>2</sup> Prioridad del proceso 1<sup>3</sup>

<sup>1</sup>.-Es el nombre que hemos dado al proceso.

<sup>2</sup>.-Es el tiempo que tarde en ejecutarse dicho proceso.

<sup>3</sup>.- Es la prioridad que tiene dicho proceso.

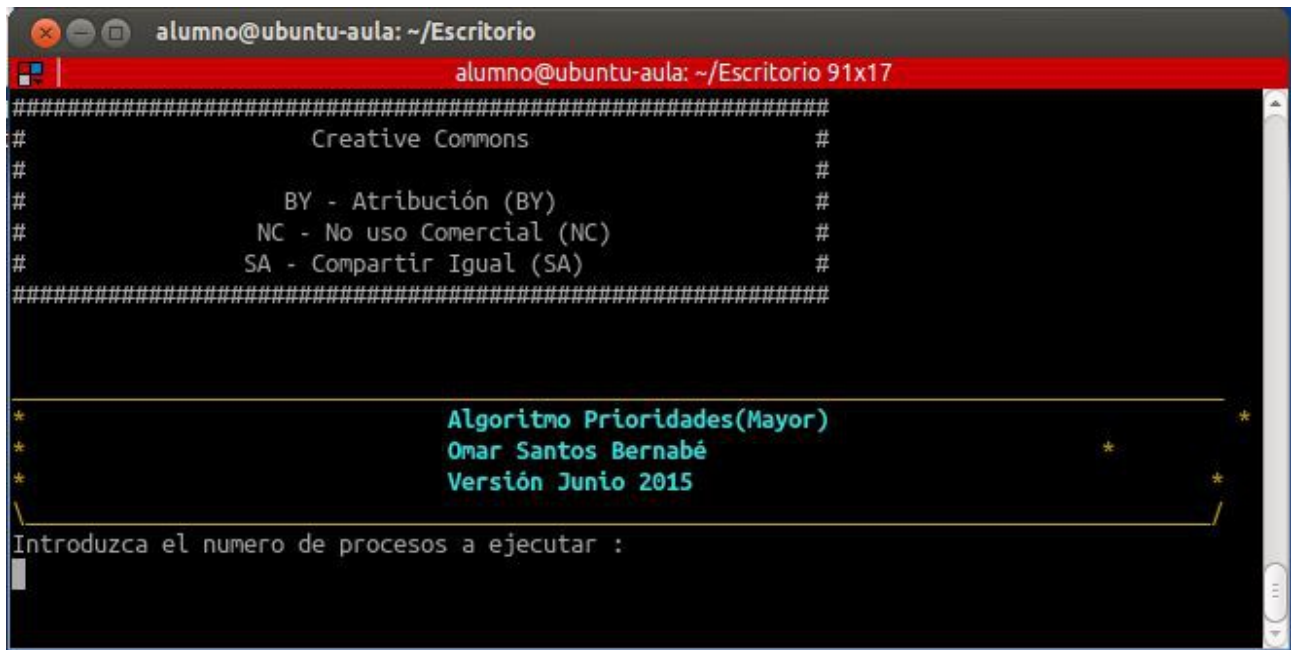
Finalmente observamos los resultados de la ejecución del algoritmo en la cual aparecen los procesos ordenados según su salida de ejecución junto con los datos del tiempo de ejecución y su prioridad, y con el tiempo de espera y el tiempo de respuesta de cada uno de los procesos. Al final aparecen los tiempos de espera medio y de retorno medio del algoritmo a partir de los datos que hemos introducido.

Una vez cerramos el fichero de texto se termina la ejecución del algoritmo.

## Ejercicio1-Mayor

A continuación veremos mediante volcados de pantalla la ejecución del algoritmo con los datos del primer ejercicio realizado a mano:

Lo primero de todo que se tiene que realizar es ir a la carpeta o directorio en el que se encuentre el algoritmo y una vez estemos ahí situados utilizaremos el comando “./PlanificacionPRIORIDADES\_MAYOR.sh” para ejecutarlo. Una vez hemos ejecutado el algoritmo lo primero que nos aparecerá sera esto.



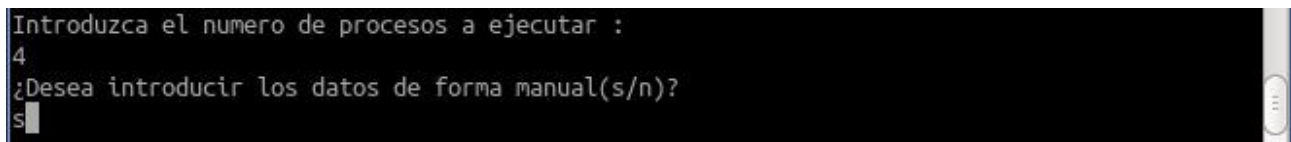
```
alumno@ubuntu-aula: ~/Escritorio
alumno@ubuntu-aula: ~/Escritorio 91x17
#####
#           Creative Commons           #
#                                     #
#           BY - Atribución (BY)      #
#           NC - No uso Comercial (NC) #
#           SA - Compartir Igual (SA)  #
#####

*           Algoritmo Prioridades(Mayor)           *
*           Omar Santos Bernabé                     *
*           Versión Junio 2015                       *
\_____/

Introduzca el numero de procesos a ejecutar :

```

Aparece una licencia de creative commons que indica que es una versión nuestra y que no puede ser modificada sin autorización y que no puede usarse con un fin comercial. A continuación nos aparece la cabecera del algoritmo en la que podemos encontrar el nombre del algoritmo, el nombre del alumno que lo ha modificado por última vez y la versión a la que corresponde el algoritmo, en este caso la última versión realizada que es la de Junio del 2015.



```
Introduzca el numero de procesos a ejecutar :
4
¿Desea introducir los datos de forma manual(s/n)?
s

```

Como se podía ver en la imagen anterior al final nos pedía introducir el número de procesos que queremos ejecutar. Introducimos un total de cuatro procesos para que coincida con el número de procesos que hemos seleccionado en el primer ejercicio que hemos hecho de forma manual. A continuación nos pregunta si queremos introducir los datos de forma manual. En el caso de introducir como respuesta una s nos pedirá como se vera a continuación paso a paso cada uno de los datos necesarios

para cada uno de los procesos. Si por el contrario introducimos una n, como se verá en el próximo ejemplo, cogerá los datos del fichero datosEntradaPredeterminados.txt en el que hay datos para varios procesos.

Proceso	T. Ejecución	Prioridad
P1	3	1
P2	7	2
P3	8	1
P4	4	

Prioridad del proceso 3  
2

Lo primero que introducíamos como se ve en la imagen anterior era el nombre para el primer proceso y una vez introducido nos aparecía la siguiente tabla en la que podemos ver de forma mas clara los datos que ya han sido introducidos para así evatar introducir los mismos tiempos de llegada y de ejecución para distintos procesos, lo que sería inútil introducir varios procesos con los mismos datos puesto que de ese modo no conseguiríamos ver bien el funcionamiento de este algoritmo.

Como se puede observar se introducen los datos de forma ordenada cada uno para su procesos, y el algoritmo nos va pidiendo que introduzcamos primero el nombre que queremos para el proceso, a continuación el tiempo de llegada, el instante en el que dicho proceso llega para ejecutarse y finalmente el tiempo de ejecución que indica el tiempo que estará ejecutandose dicho proceso. Se realiza esto para cada uno de los procesos, en este caso para cuatro procesos que es el número de procesos que queríamos ejecutar y una vez hemos introducido los datos paraa todos los procesos el algoritmo se dispone a ejecutarse.

```
Proceso: P1 Tiempo de ejecución: 3 Prioridad de Proceso: 1
Proceso: P2 Tiempo de ejecución: 7 Prioridad de Proceso: 2
Proceso: P3 Tiempo de ejecución: 8 Prioridad de Proceso: 1
Proceso: P4 Tiempo de ejecución: 4 Prioridad de Proceso: 2
```

Lo primero que aparece una vez hemos terminado de introducir los datos de todos los procesos es una pequeña lista en la que podemos encontrarnos el nombre del proceso, el tiempo de llegada y el tiempo de ejecución de cada uno de los procesos que hemos introducido antes. De seguido podemos ver el orden de ejecución de los diferentes procesos en una pequeña tabla que contiene los procesos, su tiempo de llegada, su tiempo de ejecución, su tiempo de espera y su tiempo de respuesta.

Proceso	Ejecución	Prioridad	Espera	Respuesta
* P2 *	* 7 *	* 2 *	* 0 *	* 7 *
* P4 *	* 4 *	* 2 *	* 7 *	* 11 *
* P3 *	* 8 *	* 1 *	* 11 *	* 19 *
* P1 *	* 3 *	* 1 *	* 19 *	* 22 *
* T.espera medio: 9 - * T.retorno medio: 14				

Como podemos observar el primer proceso en ejecutarse es el proceso P2 que tiene prioridad dos y como el otro proceso con prioridad dos, P4, tiene menor tiempo de ejecución comienza P2 a ejecutarse, por lo que tiene cero unidades de tiempo de espera y como su ejecución es de siete unidades de tiempo, su tiempo de respuesta es de siete unidades. En el momento que termina el proceso P4 comienza la ejecución. Su tiempo de ejecución era de cuatro unidades de tiempo y tenía una espera de siete unidades que se debía a que el proceso P2 se había ejecutado antes que este, por lo que su tiempo de respuesta final era de once unidades de tiempo ya que se suman su ejecución y su tiempo de espera. El siguiente proceso que entra a ejecutarse una vez termina de ejecutarse el proceso P4 es el proceso P3, que tiene prioridad uno y comienza a ejecutarse antes del proceso P1 que también tiene prioridad uno porque su tiempo de ejecución es mayor que el de P1. Por ese motivo comienza la ejecución de P3 que tiene un tiempo de ejecución de ocho unidades de tiempo y que tiene un tiempo de espera de once unidades, por lo que su tiempo final de respuesta es de diecinueve unidades de tiempo que sale de sumar su espera con su tiempo de ejecución. Finalmente se ejecuta el proceso P1 que tenía un tiempo de ejecución de tres unidades de tiempo y una espera de diecinueve por lo que su tiempo de respuesta final era de veintidós unidades de tiempo. Con el resultado de realizar el algoritmo podemos conocer los tiempos de espera medios que en este caso es de nueve unidades de tiempo y el tiempo de retorno medio que es de catorce unidades para este ejemplo.

¿Quieres abrir el informe? ([s],n): s

Por último se nos da la opción de abrir o no un informe con el resultado de la ejecución del algoritmo. A la hora de abrir el informe nos da dos opciones, introducir una s para abrirlo, que de igual forma funciona introduciendo una S o simplemente pulsando intro con el hueco vacío, o una n o N para no abrirlo y terminar la ejecución del algoritmo. En el caso de abrir el informe se nos abrirá un archivo txt como el siguiente.



```
informePrioridadMayor.txt X
#####
#                               #
#           Creative Commons     #
#                               #
#           BY - Atribución (BY) #
#           NC - No uso Comercial (NC) #
#           SA - Compartir Igual (SA) #
#                               #
#####

#####
#                               #
#           INFORME DE PRÁCTICA  #
#           GESTIÓN DE PROCESOS  #
#           -----              #
#           Nuevos alumnos:      #
#           Alumnos: Omar Santos Bernabé #
#           Sistemas Operativos 2º Semestre #
#           Grado en ingeniería informática (2015-2016) #
#                               #
#####

Proceso: P1 Tiempo de ejecución: 3 Prioridad de Proceso: 1
Proceso: P2 Tiempo de ejecución: 7 Prioridad de Proceso: 2
Proceso: P3 Tiempo de ejecución: 8 Prioridad de Proceso: 1
Proceso: P4 Tiempo de ejecución: 4 Prioridad de Proceso: 2
```

En el informe se puede ver de nuevo al principio la licencia del creative commons que como he explicado antes nos indica que el algoritmo no es de uso comercial y que no se debe modificar, sino que debe entregarse como se a recibido.

A continuación se observa una cabecera que indica el nombre del último alumno que ha realizado esta versión del algoritmo y el año o curso en el cual se ha realizado dicha versión.

Podemos observar los datos que se han introducido para la realización de este ejercicio en el orden que han sido introducidos. Los datos de muestran de la siguiente forma:

Proceso P1<sup>1</sup> Tiempo de ejecución 3<sup>2</sup> Prioridad del proceso 1<sup>3</sup>

<sup>1</sup>.-Es el nombre que hemos dado al proceso.

<sup>2</sup>.-Es el tiempo que tarde en ejecutarse dicho proceso.

<sup>3</sup>.- Es la prioridad que tiene dicho proceso.

	Proceso		Ejecución		Prioridad		Espera		Respuesta	
*	P2	*	7	*	2	*	0	*	7	*
*	P4	*	4	*	2	*	7	*	11	*
*	P3	*	8	*	1	*	11	*	19	*
*	P1	*	3	*	1	*	19	*	22	*
* T.espera medio: 9 - * T.retorno medio: 14										

Finalmente observamos los resultados de la ejecución del algoritmo en la cual aparecen los procesos ordenados según su salida de ejecución junto con los datos del tiempo de ejecución y su prioridad, y con el tiempo de espera y el tiempo de respuesta de cada uno de los procesos. Al final aparecen los tiempos de espera medio y de retorno medio del algoritmo a partir de los datos que hemos introducido.

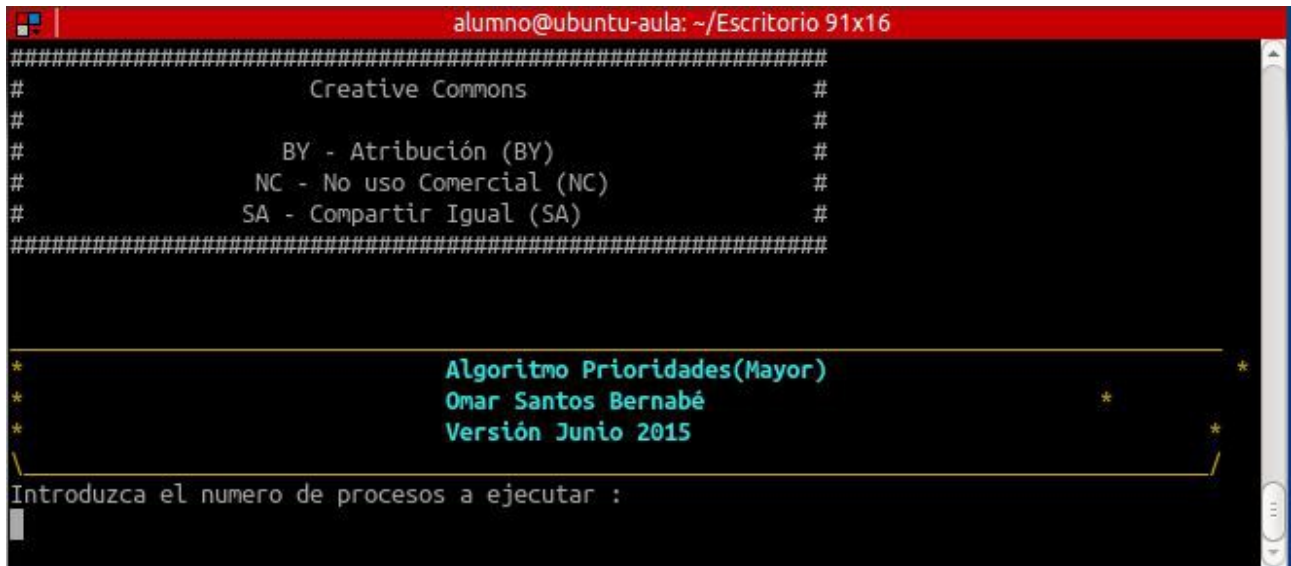
Una vez cerramos el fichero de texto se termina la ejecución del algoritmo.



## Ejercicio2-Mayor

A continuación veremos mediante volcados de pantalla la ejecución del algoritmo con los datos del primer ejercicio realizado a mano:

Lo primero de todo que se tiene que realizar es ir a la carpeta o directorio en el que se encuentre el algoritmo y una vez estemos ahí situados utilizaremos el comando “./PlanificacionPRIORIDADES\_MAYOR.sh” para ejecutarlo. Una vez hemos ejecutado el algoritmo lo primero que nos aparecerá sera esto.

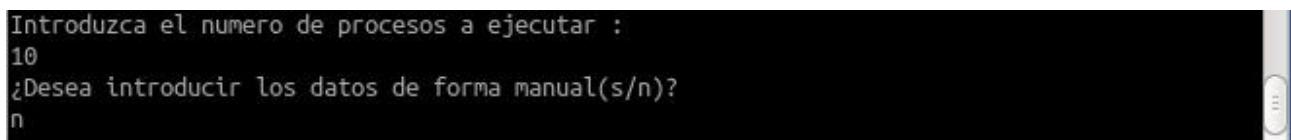


```
alumno@ubuntu-aula: ~/Escritorio 91x16
#####
#           Creative Commons           #
#                                     #
#           BY - Atribución (BY)      #
#           NC - No uso Comercial (NC) #
#           SA - Compartir Igual (SA)  #
#####

*                                     *
*           Algoritmo Prioridades(Mayor)           *
*           Omar Santos Bernabé                     *
*           Versión Junio 2015                       *
*                                     *
Introduzca el numero de procesos a ejecutar :

```

Aparece una licencia de creative commons que indica que es una versión nuestra y que no puede ser modificada sin autorización y que no puede usarse con un fin comercial. A continuación nos aparece la cabecera del algoritmo en la que podemos encontrar el nombre del algoritmo, el nombre del alumno que lo ha modificado por última vez y la versión a la que corresponde el algoritmo, en este caso la última versión realizada que es la de Junio del 2015.



```
Introduzca el numero de procesos a ejecutar :
10
¿Desea introducir los datos de forma manual(s/n)?
n

```

Como se podía ver en la imagen anterior al final nos pedía introducir el número de procesos que queremos ejecutar. Introducimos un total de 10 procesos para que coincida con el número de procesos que hemos seleccionado en el segundo ejercicio que hemos hecho de forma manual. A continuación nos pregunta si queremos introducir los datos de forma manual. Introducimos una n para indicar que queremos obtener los datos del fichero de datosEntradaPredeterminados.txt en el que hay datos para un alto número de procesos.

```
Introduzca el nombre del proceso 0:
Tiempo De ejecucion del proceso 0:
Prioridad del proceso 0:
Introduzca el nombre del proceso 1:
Tiempo De ejecucion del proceso 1:
Prioridad del proceso 1:
Introduzca el nombre del proceso 2:
Tiempo De ejecucion del proceso 2:
Prioridad del proceso 2:
Introduzca el nombre del proceso 3:
Tiempo De ejecucion del proceso 3:
Prioridad del proceso 3:
Introduzca el nombre del proceso 4:
Tiempo De ejecucion del proceso 4:
Prioridad del proceso 4:
Introduzca el nombre del proceso 5:
Tiempo De ejecucion del proceso 5:
Prioridad del proceso 5:
Introduzca el nombre del proceso 6:
Tiempo De ejecucion del proceso 6:
Prioridad del proceso 6:
Introduzca el nombre del proceso 7:
Tiempo De ejecucion del proceso 7:
Prioridad del proceso 7:
Introduzca el nombre del proceso 8:
Tiempo De ejecucion del proceso 8:
Prioridad del proceso 8:
Introduzca el nombre del proceso 9:
Tiempo De ejecucion del proceso 9:
Prioridad del proceso 9:
```

Como se puede observar en la imagen el propio algoritmo va cogiendo en orden los datos que necesita para los diez procesos que vamos a ejecutar. Los datos se recogen en el mismo orden que se pedían de forma manual, primero el nombre que queremos dar al proceso, después el tiempo de llegada de dicho proceso y finalmente el tiempo de ejecución para ese proceso. Se realiza ese paso hasta recoger los datos para los diez procesos que queremos.

Se introducen los datos de forma ordenada cada uno para su procesos, y el algoritmo nos va pidiendo que introduzcamos primero el nombre que queremos para el proceso, a continuación el tiempo de ejecución, que indica el tiempo que estará ejecutándose dicho proceso y finalmente su prioridad. Se realiza esto para cada uno de los procesos, en este caso para diez procesos que es el número de procesos que queríamos ejecutar y una vez ha recogido los datos del fichero para todos los procesos el algoritmo se dispone a ejecutarse.

```

Proceso: P1 Tiempo de ejecución: 10 Prioridad del Proceso: 3
Proceso: P2 Tiempo de ejecución: 1 Prioridad del Proceso: 1
Proceso: P3 Tiempo de ejecución: 2 Prioridad del Proceso: 3
Proceso: P4 Tiempo de ejecución: 1 Prioridad del Proceso: 4
Proceso: P5 Tiempo de ejecución: 5 Prioridad del Proceso: 2
Proceso: P6 Tiempo de ejecución: 2 Prioridad del Proceso: 7
Proceso: P7 Tiempo de ejecución: 1 Prioridad del Proceso: 12
Proceso: P8 Tiempo de ejecución: 4 Prioridad del Proceso: 1
Proceso: P9 Tiempo de ejecución: 2 Prioridad del Proceso: 1
Proceso: P10 Tiempo de ejecución: 2 Prioridad del Proceso: 2

```

Lo primero que aparece una vez hemos terminado de introducir los datos de todos los procesos es una pequeña lista en la que podemos encontrarnos el nombre del proceso, el tiempo de ejecución y la prioridad de cada uno de los procesos que hemos introducido antes. De seguido podemos ver el orden de ejecución de los diferentes procesos en una pequeña tabla que contiene los procesos, su tiempo de llegada, su tiempo de ejecución, su tiempo de espera y su tiempo de respuesta.

Proceso	Ejecución	Prioridad	Espera	Respuesta
* P7 *	1	12	0	1
* P6 *	2	7	1	3
* P4 *	1	4	3	4
* P1 *	10	3	4	14
* P3 *	2	3	14	16
* P5 *	5	2	16	21
* P10 *	2	2	21	23
* P8 *	4	1	23	27
* P9 *	2	1	27	29
* P2 *	1	1	29	30
* T.espera medio: 13 - * T.retorno medio: 16				

El orden ejecución de los procesos será en función de la prioridad asignada a los procesos. En este caso al seguir una prioridad mayor el primer proceso en ejecutarse será el proceso P7 durante una ráfaga de CPU por lo que su tiempo de respuesta es también de una unidad de tiempo puesto que su tiempo de espera es de cero. Se ejecuta a continuación el proceso P6 por ser el siguiente con mayor prioridad con un tiempo de espera de una unidad de tiempo debido a la ejecución de P7 antes, y según termina sus dos unidades de tiempo de ejecución dejando un tiempo de respuesta final de tres unidades de tiempo se ejecuta el proceso P4 durante una unidad de tiempo por lo que sumado a su tiempo de espera que es de tres unidades deja un tiempo de respuesta de cuatro unidades de tiempo. A continuación se ejecuta el proceso P1 que de los que tienen mayor prioridad es el que mayor tiempo de ejecución tiene, ejecutándose durante diez unidades de tiempo y dejando un tiempo de respuesta final de catorce unidades. Según termina se ejecuta el proceso P3 que tiene su misma prioridad pero menor tiempo de ejecución, como tiene un tiempo de espera de catorce unidades de tiempo y su tiempo de ejecución es de dos unidades su tiempo de respuesta es de dieciséis unidades de tiempo. A continuación se ejecuta el proceso P5 durante cinco ráfagas de tiempo porque tiene la misma prioridad que el proceso P10

pero mayor tiempo de ejecución, dejando un tiempo de respuesta de veintiuna unidades de tiempo. El siguiente proceso que se ejecuta es el proceso P10 durante dos unidades de tiempo que sumado a sus veintiuna unidades de espera deja un tiempo de respuesta de veintitrés unidades de tiempo.. Por último se ejecutan los que menor prioridad tiene, y de estos el primero en ejecutarse es el proceso P8 porque tiene un mayor tiempo de ejecución que los otros dos procesos y se ejecuta durante cuatro unidades de tiempo. El siguiente es el proceso P9 durante dos ráfagas de CPU. Y finalmente se ejecuta el proceso P2 durante una unidad de tiempo dejando un tiempo de respuesta de treinta unidades que surgen de sumar su unidad de tiempo de ejecución con las veintinueve que tenía en tiempo de espera.

```
¿Quieres abrir el informe? ([s],n): s
```

Por último se nos da la opción de abrir o no un informe con el resultado de la ejecución del algoritmo. A la hora de abrir el informe nos da dos opciones, introducir una s para abrirlo, que de igual forma funciona introduciendo una S o simplemente pulsando intro con el hueco vacío, o una n o N para no abrirlo y terminar la ejecución del algoritmo. En el caso de abrir el informe se nos abrirá un archivo txt como el siguiente.

```
informePrioridadMayor.txt X
#####
#                               #
#           Creative Commons     #
#                               #
#           BY - Atribución (BY) #
#           NC - No uso Comercial (NC) #
#           SA - Compartir Igual (SA) #
#                               #
#####

#                               #
#           INFORME DE PRÁCTICA   #
#           GESTIÓN DE PROCESOS  #
#           -----              #
#           Nuevos alumnos:      #
#           Alumnos: Omar Santos Bernabé #
#           Sistemas Operativos 2º Semestre #
#           Grado en ingeniería informática (2015-2016) #
#                               #
#####

Proceso: P1 Tiempo de ejecución: 10 Prioridad de Proceso: 3
Proceso: P2 Tiempo de ejecución: 1 Prioridad de Proceso: 1
Proceso: P3 Tiempo de ejecución: 2 Prioridad de Proceso: 3
Proceso: P4 Tiempo de ejecución: 1 Prioridad de Proceso: 4
Proceso: P5 Tiempo de ejecución: 5 Prioridad de Proceso: 2
Proceso: P6 Tiempo de ejecución: 2 Prioridad de Proceso: 7
```



Proceso: P7 Tiempo de ejecución: 1 Prioridad de Proceso: 12  
 Proceso: P8 Tiempo de ejecución: 4 Prioridad de Proceso: 1  
 Proceso: P9 Tiempo de ejecución: 2 Prioridad de Proceso: 1  
 Proceso: P10 Tiempo de ejecución: 2 Prioridad de Proceso: 2

	Proceso		Ejecución		Prioridad		Espera		Respuesta	
*	P7	*	1	*	12	*	0	*	1	*
*	P6	*	2	*	7	*	1	*	3	*
*	P4	*	1	*	4	*	3	*	4	*
*	P1	*	10	*	3	*	4	*	14	*
*	P3	*	2	*	3	*	14	*	16	*
*	P5	*	5	*	2	*	16	*	21	*
*	P10	*	2	*	2	*	21	*	23	*
*	P8	*	4	*	1	*	23	*	27	*
*	P9	*	2	*	1	*	27	*	29	*
*	P2	*	1	*	1	*	29	*	30	*

\* T.espera medio: 13 - \* T.retorno medio: 16

En el informe se puede ver de nuevo al principio la licencia del creative commons que como he explicado antes nos indica que el algoritmo no es de uso comercial y que no se debe modificar, sino que debe entregarse como se a recibido.

A continuación se observa una cabecera que indica el nombre del último alumno que ha realizado esta versión del algoritmo y el año o curso en el cual se ha realizado dicha versión.

Podemos observar los datos que se han introducido para la realización de este ejercicio en el orden que han sido introducidos. Los datos de muestran de la siguiente forma:

Proceso P1<sup>1</sup> Tiempo de ejecución 3<sup>2</sup> Prioridad del proceso 1<sup>3</sup>

<sup>1</sup>.-Es el nombre que hemos dado al proceso.

<sup>2</sup>.-Es el tiempo que tarde en ejecutarse dicho proceso.

<sup>3</sup>.- Es la prioridad que tiene dicho proceso.

Finalmente observamos los resultados de la ejecución del algoritmo en la cual aparecen los procesos ordenados según su salida de ejecución junto con los datos del tiempo de ejecución y su prioridad, y con el tiempo de espera y el tiempo de respuesta de cada uno de los procesos. Al final aparecen los tiempos de espera medio y de retorno medio del algoritmo a partir de los datos que hemos introducido.

Una vez cerramos el fichero de texto se termina la ejecución del algoritmo.

### 3. Datos de interés sobre el código

```
function ComprobarNombreProceso {  
    palabra=`echo $1 $2 | wc -w` #cuento el número de palabras  
    if [ $palabra -ne 1 ] #si es distinto de 1  
    then  
        echo -e "${red}Nombre De Proceso No Valido${NC}"  
        exit  
    fi  
}
```

En esta imagen podemos ver como en una función comprobamos que el número de palabras introducidas sea el adecuado. Esta función se la llama donde se piden los nombres de los procesos para comprobar que el nombre introducido sea válido, es decir, que el nombre esté formado por solo una palabra. En el caso de que el número de palabras introducidas sea distinto e uno nos saldría un mensaje de error terminando la ejecución del script.

```
echo "Introduzca el nombre del proceso $p:"  
nombre=`cat datosEntradaPredefinidos.txt | cut -d ";" -f $pp` #  
ComprobarNombreProceso $nombre #comprobamos que la entrada leída  
proceso[$p]=$nombre; #añadimos al vector de los proce  
pp=`expr $pp + 1`  
echo "Tiempo De ejecución del proceso $p:"  
tiemp=`cat datosEntradaPredefinidos.txt | cut -d ";" -f $pp`  
tiempo[$p]=$tiemp;  
pp=`expr $pp + 1`  
echo "Prioridad del proceso $p:"  
priorida=`cat datosEntradaPredefinidos.txt | cut -d ";" -f $pp`  
prioridad[$p]=$priorida;
```

Como podemos ver en esta otra imagen, en el momento de pedir el nombre de cada proceso, en este caso por fichero, se coge como dato lo que esta justo antes del punto y coma de separación de los distintos datos de los procesos. Una vez se a cogido ese dato se hace la llamada a la función anterior enviando el nombre que se ha cogido para comprobar que es un nombre correcto, y en el caso de serlo se añade a la lista de procesos.