

Practica1

1

Generated by Doxygen 1.8.7

Thu Feb 25 2016 02:10:49

Contents

1	File Index	1
1.1	File List	1
2	File Documentation	3
2.1	ejercicio4a.c File Reference	3
2.1.1	Detailed Description	3
2.1.2	Function Documentation	4
2.1.2.1	main	4
2.2	ejercicio4b.c File Reference	4
2.2.1	Detailed Description	4
2.2.2	Function Documentation	5
2.2.2.1	main	5
2.3	ejercicio5a.c File Reference	5
2.3.1	Detailed Description	5
2.3.2	Function Documentation	6
2.3.2.1	main	6
2.4	ejercicio5b.c File Reference	6
2.4.1	Detailed Description	6
2.4.2	Function Documentation	7
2.4.2.1	main	7
2.5	ejercicio6.c File Reference	7
2.5.1	Detailed Description	7
2.5.2	Function Documentation	8
2.5.2.1	main	8
2.6	ejercicio8a.c File Reference	8
2.6.1	Detailed Description	8
2.6.2	Function Documentation	8
2.6.2.1	main	8
2.7	ejercicio8b.c File Reference	9
2.7.1	Detailed Description	9
2.7.2	Function Documentation	9

2.7.2.1	main	9
2.8	ejercicio9.c File Reference	10
2.8.1	Detailed Description	10
2.8.2	Function Documentation	10
2.8.2.1	main	10

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

ejercicio4a.c		
Modulo del apartado a) del ejercicio 4		3
ejercicio4b.c		
Modulo del apartado b) del ejercicio 4		4
ejercicio5a.c		
Modulo del apartado a) del ejercicio 5		5
ejercicio5b.c		
Modulo del apartado b) del ejercicio 5		6
ejercicio6.c		
Ejercicio 6		7
ejercicio8a.c		
Modulo del apartado a) del ejercicio 8		8
ejercicio8b.c		
Modulo del apartado b) del ejercicio 8		9
ejercicio9.c		
Ejercicio 9. Contiene un programa que lanza 2 hijos, estos lanzan a a su vez 2 hijos mas. Despues mediante tuberias, el padre envia un mensaje a sus nietos. Estos lo imprimen por pantalla junto con el PID del proceso por donde lo recibieron y mandan una respuesta. Estas respuestas son recibidas por el padre, que las imprime por pantalla junto con el PID del nieto que las mando		10

Chapter 2

File Documentation

2.1 ejercicio4a.c File Reference

Modulo del apartado a) del ejercicio 4.

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

Macros

- `#define NUM_PROC 3`

Functions

- `int main (void)`
main del programa del ejercicio 4a

2.1.1 Detailed Description

Modulo del apartado a) del ejercicio 4.

Contiene un programa que lanza varios hijos y en cada `fork()` el proceso hijo imprime en la salida estandar su `pid` y el de su padre

Author

Pablo Marcos Manchon pablo.marcosm@estudiante.uam.es
David Nevado Catalan david.nevadoc@estudiante.uam.es

Version

1.0

Date

23-02-2016

Definition in file [ejercicio4a.c](#).

2.1.2 Function Documentation

2.1.2.1 `int main (void)`

main del programa del ejercicio 4a

Returns

EXIT_SUCCESS si fue todo bien o EXIT_FAILURE en caso contrario

Definition at line 26 of file ejercicio4a.c.

2.2 `ejercicio4b.c` File Reference

Modulo del apartado b) del ejercicio 4.

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

Macros

- `#define NUM_PROC 3`

Functions

- `int main (void)`
main del programa del ejercicio 4b

2.2.1 Detailed Description

Modulo del apartado b) del ejercicio 4.

Contiene un programa que lanza varios hijos y en cada `fork()` el proceso hijo imprime en la salida estandar su pid y el de su padre, y realiza un `wait()` antes de finalizar

Author

Pablo Marcos Manchon pablo.marcosm@estudiante.uam.es
David Nevado Catalan david.nevadoc@estudiante.uam.es

Version

1.0

Date

23-02-2016

Definition in file [ejercicio4b.c](#).

2.2.2 Function Documentation

2.2.2.1 int main (void)

main del programa del ejercicio 4b

Returns

EXIT_SUCCESS si fue todo bien o EXIT_FAILURE en caso contrario

Definition at line 26 of file ejercicio4b.c.

2.3 ejercicio5a.c File Reference

Modulo del apartado a) del ejercicio 5.

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

Macros

- #define NUM_PROC 3

Functions

- int [main](#) (void)

main del programa del ejercicio 5a

2.3.1 Detailed Description

Modulo del apartado a) del ejercicio 5.

Se genera secuencialmente una serie de procesos que imprimen su pid y el de su padre

Author

Pablo Marcos Manchon pablo.marcosm@estudiante.uam.es
David Nevado Catalan david.nevadoc@estudiante.uam.es

Version

1.0

Date

23-02-2016

Definition in file [ejercicio5a.c](#).

2.3.2 Function Documentation

2.3.2.1 `int main (void)`

main del programa del ejercicio 5a

Returns

EXIT_SUCCESS si fue todo bien o EXIT_FAILURE en caso contrario

Definition at line 24 of file ejercicio5a.c.

2.4 ejercicio5b.c File Reference

Modulo del apartado b) del ejercicio 5.

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

Macros

- `#define NUM_PROC 3`

Functions

- `int main (void)`
main del programa del ejercicio 4b

2.4.1 Detailed Description

Modulo del apartado b) del ejercicio 5.

Contiene un programa que lanza varios hijos esperando hasta que termine la ejecucion de todos ellos Cada hijo imprime su pid y el de su padre

Author

Pablo Marcos Manchon pablo.marcosm@estudiante.uam.es
David Nevado Catalan david.nevadoc@estudiante.uam.es

Version

1.0

Date

23-02-2016

Definition in file [ejercicio5b.c](#).

2.4.2 Function Documentation

2.4.2.1 int main (void)

main del programa del ejercicio 4b

Returns

EXIT_SUCCESS si fue todo bien o EXIT_FAILURE en caso contrario

Definition at line 25 of file ejercicio5b.c.

2.5 ejercicio6.c File Reference

Ejercicio 6.

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

Macros

- #define **TAM** 30

Functions

- int [main](#) (void)

main del programa del ejercicio 6

2.5.1 Detailed Description

Ejercicio 6.

Programa que comprueba que unos procesos padre e hijo no comparten variables

Author

Pablo Marcos Manchon pablo.marcosm@estudiante.uam.es
David Nevado Catalan david.nevadoc@estudiante.uam.es

Version

1.0

Date

23-02-2016

Definition in file [ejercicio6.c](#).

2.5.2 Function Documentation

2.5.2.1 int main (void)

main del programa del ejercicio 6

Returns

EXIT_SUCCESS si fue todo bien o EXIT_FAILURE en caso contrario

Definition at line 28 of file ejercicio6.c.

2.6 ejercicio8a.c File Reference

Modulo del apartado a) del ejercicio 8.

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

Functions

- int [main](#) (void)
main del programa del ejercicio 8a

2.6.1 Detailed Description

Modulo del apartado a) del ejercicio 8.

Programa que devuelve el tamaño de su ejecutable mediante el comando du

Author

Pablo Marcos Manchon pablo.marcosm@estudiante.uam.es
David Nevado Catalan david.nevadoc@estudiante.uam.es

Version

1.0

Date

23-02-2016

Definition in file [ejercicio8a.c](#).

2.6.2 Function Documentation

2.6.2.1 int main (void)

main del programa del ejercicio 8a

Returns

devuelve EXIT_FAILURE en caso de fallo

Definition at line 24 of file ejercicio8a.c.

2.7 ejercicio8b.c File Reference

Modulo del apartado b) del ejercicio 8.

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

Macros

- `#define TAM 40`

Functions

- `int main (void)`
main del programa del ejercicio 8b

2.7.1 Detailed Description

Modulo del apartado b) del ejercicio 8.

Programa que solicita al usuario un comando para ejecutar y despues ofrece la posibilidad de ejecutarlo en primer o segundo plano

Author

Pablo Marcos Manchon pablo.marcosm@estudiante.uam.es
David Nevado Catalan david.nevadoc@estudiante.uam.es

Version

1.0

Date

23-02-2016

Definition in file [ejercicio8b.c](#).

2.7.2 Function Documentation

2.7.2.1 `int main (void)`

main del programa del ejercicio 8b

Returns

Devuelve `EXIT_FAILURE` en caso de fallo, y `EXIT_SUCCESS` si se ejecuta el comando en segundo plano y termina con exito. Los parametros junto al comando que se introduzcan son ignorados.

Definition at line 31 of file [ejercicio8b.c](#).

2.8 ejercicio9.c File Reference

Ejercicio 9. Contiene un programa que lanza 2 hijos, estos lanzan a su vez 2 hijos mas. Despues mediante tuberias, el padre envia un mensaje a sus nietos. Estos lo imprimen por pantalla junto con el PID del proceso por donde lo recibieron y mandan una respuesta. Estas respuestas son recibidas por el padre, que las imprime por pantalla junto con el PID del nieto que las mando.

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

Macros

- `#define TAM 150`

Functions

- `int main (void)`

main del programa del ejercicio 9

2.8.1 Detailed Description

Ejercicio 9. Contiene un programa que lanza 2 hijos, estos lanzan a su vez 2 hijos mas. Despues mediante tuberias, el padre envia un mensaje a sus nietos. Estos lo imprimen por pantalla junto con el PID del proceso por donde lo recibieron y mandan una respuesta. Estas respuestas son recibidas por el padre, que las imprime por pantalla junto con el PID del nieto que las mando.

Author

David Nevado Catalan david.nevadoc@estudiante.uam.es
Pablo Marcos Manchon pablo.marcosm@estudiante.uam.es

Version

1.0

Date

23-02-2016

Definition in file [ejercicio9.c](#).

2.8.2 Function Documentation

2.8.2.1 `int main (void)`

main del programa del ejercicio 9

Returns

EXIT_SUCCESS si fue todo bien o EXIT_FAILURE en caso contrario

Definition at line 30 of file ejercicio9.c.