

Práctica 1

Generado por Doxygen 1.8.11

Índice general

| | | |
|----------|--|----------|
| 1 | Índice de estructura de datos | 1 |
| 1.1 | Estructura de datos | 1 |
| 2 | Índice de archivos | 3 |
| 2.1 | Lista de archivos | 3 |
| 3 | Documentación de las estructuras de datos | 5 |
| 3.1 | Referencia de la Estructura Estructura | 5 |
| 3.1.1 | Descripción detallada | 5 |
| 3.1.2 | Documentación de los campos | 5 |
| 3.1.2.1 | cadena | 5 |
| 3.1.2.2 | num | 5 |
| 3.2 | Referencia de la Estructura Matriz | 6 |
| 3.2.1 | Descripción detallada | 6 |
| 3.2.2 | Documentación de los campos | 6 |
| 3.2.2.1 | dim | 6 |
| 3.2.2.2 | fila | 6 |
| 3.2.2.3 | hilo | 6 |
| 3.2.2.4 | m | 6 |
| 3.2.2.5 | mult | 6 |

| | |
|---|----------|
| 4 Documentación de archivos | 7 |
| 4.1 Referencia del Archivo ejercicio12a.c | 7 |
| 4.2 Referencia del Archivo ejercicio12b.c | 7 |
| 4.2.1 Documentación de los 'defines' | 8 |
| 4.2.1.1 TAM_HILO | 8 |
| 4.2.2 Documentación de las funciones | 8 |
| 4.2.2.1 main(int argc, char **argv) | 8 |
| 4.2.2.2 primerosNPrimos(int *N) | 8 |
| 4.3 Referencia del Archivo ejercicio13.c | 8 |
| 4.3.1 Descripción detallada | 9 |
| 4.3.2 Documentación de los 'defines' | 9 |
| 4.3.2.1 MAX_TAM | 9 |
| 4.3.2.2 NUM_HILOS | 10 |
| 4.3.2.3 RETARDO | 10 |
| 4.3.3 Documentación de las funciones | 10 |
| 4.3.3.1 main(void) | 10 |
| 4.3.3.2 multiplica_matriz(Matriz *m) | 10 |
| 4.4 Referencia del Archivo ejercicio4.c | 10 |
| 4.4.1 Descripción detallada | 11 |
| 4.4.2 Documentación de los 'defines' | 11 |
| 4.4.2.1 NUM_PROC | 11 |
| 4.4.3 Documentación de las funciones | 11 |
| 4.4.3.1 main(void) | 11 |
| 4.5 Referencia del Archivo ejercicio4b.c | 11 |
| 4.5.1 Descripción detallada | 12 |
| 4.5.2 Documentación de los 'defines' | 12 |
| 4.5.2.1 NUM_PROC | 12 |
| 4.5.3 Documentación de las funciones | 12 |
| 4.5.3.1 main(void) | 12 |
| 4.6 Referencia del Archivo ejercicio5a.c | 12 |

| | | |
|----------|--|----|
| 4.6.1 | Descripción detallada | 13 |
| 4.6.2 | Documentación de los 'defines' | 13 |
| 4.6.2.1 | NUM_PROC | 13 |
| 4.6.3 | Documentación de las funciones | 13 |
| 4.6.3.1 | main(void) | 13 |
| 4.7 | Referencia del Archivo ejercicio5b.c | 13 |
| 4.7.1 | Descripción detallada | 14 |
| 4.7.2 | Documentación de los 'defines' | 14 |
| 4.7.2.1 | NUM_PROC | 14 |
| 4.7.3 | Documentación de las funciones | 14 |
| 4.7.3.1 | main(void) | 14 |
| 4.8 | Referencia del Archivo ejercicio6.c | 14 |
| 4.8.1 | Descripción detallada | 15 |
| 4.8.2 | Documentación de las funciones | 15 |
| 4.8.2.1 | main(void) | 15 |
| 4.9 | Referencia del Archivo ejercicio8.c | 15 |
| 4.9.1 | Descripción detallada | 16 |
| 4.9.2 | Documentación de los 'defines' | 16 |
| 4.9.2.1 | ERROR | 16 |
| 4.9.2.2 | MAX_TAM | 16 |
| 4.9.2.3 | OK | 16 |
| 4.9.3 | Documentación de las funciones | 16 |
| 4.9.3.1 | ejecutar(char *tipo, char *programa) | 16 |
| 4.9.3.2 | main(int argc, char **argv) | 17 |
| 4.10 | Referencia del Archivo ejercicio9.c | 17 |
| 4.10.1 | Descripción detallada | 17 |
| 4.10.2 | Documentación de los 'defines' | 18 |
| 4.10.2.1 | MAX_TAM | 18 |
| 4.10.3 | Documentación de las funciones | 18 |
| 4.10.3.1 | combinatoria(double a, double b) | 18 |
| 4.10.3.2 | factorial(double a) | 18 |
| 4.10.3.3 | main(void) | 18 |
| 4.10.3.4 | sumAbs(double a, double b) | 18 |

Capítulo 1

Índice de estructura de datos

1.1. Estructura de datos

Lista de estructuras con una breve descripción:

| | | |
|------------|--|---|
| Estructura | | |
| | Estructura pedida en el enunciado | 5 |
| Matriz | | |
| | Estructura con los datos para multiplicar una matriz por un multiplicador en un hilo | 6 |

Capítulo 2

Indice de archivos

2.1. Lista de archivos

Lista de todos los archivos con descripciones breves:

| | | |
|--------------------------------|-------|----|
| ejercicio12a.c | | |
| Ejercicio 12a | | 7 |
| ejercicio12b.c | | 7 |
| ejercicio13.c | | |
| Ejercicio 13 | | 8 |
| ejercicio4.c | | |
| Ejercicio 4 | | 10 |
| ejercicio4b.c | | |
| Ejercicio4b | | 11 |
| ejercicio5a.c | | |
| Ejercicio5a | | 12 |
| ejercicio5b.c | | |
| Ejercicio5a | | 13 |
| ejercicio6.c | | |
| Ejercicio6 | | 14 |
| ejercicio8.c | | |
| Ejercicio8 | | 15 |
| ejercicio9.c | | |
| Ejercicio9 | | 17 |

Capítulo 3

Documentación de las estructuras de datos

3.1. Referencia de la Estructura Estructura

[Estructura](#) pedida en el enunciado.

Campos de datos

- char [cadena](#) [100]
- int [num](#)

3.1.1. Descripción detallada

[Estructura](#) pedida en el enunciado.

[Estructura](#) pedida en el enunciado

Definición en la línea 25 del archivo ejercicio12a.c.

3.1.2. Documentación de los campos

3.1.2.1. char cadena

Definición en la línea 26 del archivo ejercicio12a.c.

3.1.2.2. int num

Definición en la línea 27 del archivo ejercicio12a.c.

La documentación para esta estructura fue generada a partir de los siguientes ficheros:

- [ejercicio12a.c](#)
- [ejercicio12b.c](#)

3.2. Referencia de la Estructura Matriz

[Estructura](#) con los datos para multiplicar una matriz por un multiplicador en un hilo.

Campos de datos

- int [hilo](#)
- int [mult](#)
- int [dim](#)
- int * [fila](#)
- int [m](#) [[MAX_TAM](#)][[MAX_TAM](#)]

3.2.1. Descripción detallada

[Estructura](#) con los datos para multiplicar una matriz por un multiplicador en un hilo.

Definición en la línea 23 del archivo ejercicio13.c.

3.2.2. Documentación de los campos

3.2.2.1. int dim

Definición en la línea 26 del archivo ejercicio13.c.

3.2.2.2. int* fila

Definición en la línea 27 del archivo ejercicio13.c.

3.2.2.3. int hilo

Definición en la línea 24 del archivo ejercicio13.c.

3.2.2.4. int m[MAX_TAM][MAX_TAM]

Definición en la línea 28 del archivo ejercicio13.c.

3.2.2.5. int mult

Definición en la línea 25 del archivo ejercicio13.c.

La documentación para esta estructura fue generada a partir del siguiente fichero:

- [ejercicio13.c](#)

Capítulo 4

Documentación de archivos

4.1. Referencia del Archivo ejercicio12a.c

Ejercicio 12a.

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

Dependencia gráfica adjunta para ejercicio12a.c:

4.2. Referencia del Archivo ejercicio12b.c

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

Dependencia gráfica adjunta para ejercicio12b.c:

Estructuras de datos

- struct [Estructura](#)
[Estructura](#) pedida en el enunciado.

'defines'

- #define [TAM_HILO](#) 100
numero de hilos que queremos hacer

Funciones

- void * `primerosNPrimos` (int *N)
Calcula los primeros N primos.
- int `main` (int argc, char **argv)
Funcion main del ejercicio12b.

4.2.1. Documentación de los 'defines'

4.2.1.1. `#define TAM_HILO 100`

numero de hilos que queremos hacer

Definición en la línea 21 del archivo ejercicio12b.c.

4.2.2. Documentación de las funciones

4.2.2.1. `int main (int argc, char ** argv)`

Funcion main del ejercicio12b.

Programa que calcula el tiempo que tarda en calcular los primeros n con hilos

Parámetros

| | |
|---------------|----------------------|
| <i>numero</i> | de primos a calcular |
|---------------|----------------------|

Definición en la línea 70 del archivo ejercicio12b.c.

4.2.2.2. `void* primerosNPrimos (int * N)`

Calcula los primeros N primos.

Parámetros

| | |
|---------------|----------------------|
| <i>numero</i> | de primos a calcular |
|---------------|----------------------|

Devuelve

void pues solo necesitamos calcular el tiempo

Definición en la línea 37 del archivo ejercicio12b.c.

4.3. Referencia del Archivo ejercicio13.c

Ejercicio 13.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <sys/wait.h>
#include <pthread.h>
Dependencia gráfica adjunta para ejercicio13.c:
```

Estructuras de datos

- struct [Matriz](#)

Estructura con los datos para multiplicar una matriz por un multiplicador en un hilo.

'defines'

- #define [NUM_HILOS](#) 2 /*Número de hilos del ejercicio*/
- #define [RETARDO](#) 500000 /*Tamaño del retardo en ms*/
- #define [MAX_TAM](#) 5 /*Tamaño máximo de la matriz*/

Funciones

- void * [multiplica_matriz](#) ([Matriz](#) *m)
función que multiplica la matriz m por el multiplicador mult de la estructura [Matriz](#)
- int [main](#) (void)
main del ejercicio13

4.3.1. Descripción detallada

Ejercicio 13.

Autor

Lucía Rivas Molina lucia.rivasmolina@estudiante.uam.es
Daniel Santo-Tomas Lopez daniel.santo-tomas@estudiante.uam.es

Fecha

2018/03/07

4.3.2. Documentación de los 'defines'

4.3.2.1. #define [MAX_TAM](#) 5 /*Tamaño máximo de la matriz*/

Definición en la línea 17 del archivo ejercicio13.c.

4.3.2.2. `#define NUM_HILOS 2 /*Número de hilos del ejercicio*/`

Definición en la línea 15 del archivo ejercicio13.c.

4.3.2.3. `#define RETARDO 500000 /*Tamaño del retardo en ms*/`

Definición en la línea 16 del archivo ejercicio13.c.

4.3.3. Documentación de las funciones

4.3.3.1. `int main (void)`

main del ejercicio13

Programa que multiplica dos matrices por un multiplicador con hilos

Definición en la línea 66 del archivo ejercicio13.c.

4.3.3.2. `void* multiplica_matriz (Matriz * m)`

función que multiplica la matriz m por el multiplicador mult de la estructura [Matriz](#)

Parámetros

| | |
|------------|--|
| <i>una</i> | estructura Matriz con su id de hilo, multiplicador, dimensión, array de filas y matriz |
|------------|--|

Definición en la línea 35 del archivo ejercicio13.c.

4.4. Referencia del Archivo ejercicio4.c

Ejercicio 4.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
```

Dependencia gráfica adjunta para ejercicio4.c:

'defines'

- `#define NUM_PROC 6`

Funciones

- `int main (void)`

main del ejercicio 4

4.4.1. Descripción detallada

Ejercicio 4.

Autor

Lucia Rivas Molina lucia.rivasmolina@estudiante.uam.es

Daniel Santo-Tomas Lopez daniel.santo-tomas@estudiante.uam.es

Fecha

2018/03/07

4.4.2. Documentación de los 'defines'

4.4.2.1. #define NUM_PROC 6

Definición en la línea 13 del archivo ejercicio4.c.

4.4.3. Documentación de las funciones

4.4.3.1. int main (void)

main del ejercicio 4

Ejercicio que realiza 6 forks sin waits

Definición en la línea 19 del archivo ejercicio4.c.

4.5. Referencia del Archivo ejercicio4b.c

Ejercicio4b.

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

Dependencia gráfica adjunta para ejercicio4b.c:

'defines'

- #define [NUM_PROC](#) 6

Funciones

- int [main](#) (void)

main del ejercicio4b

4.5.1. Descripción detallada

Ejercicio4b.

Autor

Lucia Rivas Molina lucia.rivasmolina@estudiante.uam.es

Daniel Santo-Tomas Lopez daniel.santo-tomas@estudiante.uam.es

Fecha

2018/03/07

4.5.2. Documentación de los 'defines'

4.5.2.1. #define NUM_PROC 6

Definición en la línea 14 del archivo ejercicio4b.c.

4.5.3. Documentación de las funciones

4.5.3.1. int main (void)

main del ejercicio4b

Ejercicio que realiza 6 forks con un wait

Definición en la línea 20 del archivo ejercicio4b.c.

4.6. Referencia del Archivo ejercicio5a.c

Ejercicio5a.

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

Dependencia gráfica adjunta para ejercicio5a.c:

'defines'

- #define [NUM_PROC](#) 6

Funciones

- int [main](#) (void)

main del Ejercicio5a

4.6.1. Descripción detallada

Ejercicio5a.

Autor

Lucia Rivas Molina lucia.rivasmolina@estudiante.uam.es

Daniel Santo-Tomas Lopez daniel.santo-tomas@estudiante.uam.es

Fecha

2018/03/07

4.6.2. Documentación de los 'defines'

4.6.2.1. #define NUM_PROC 6

Definición en la línea 13 del archivo ejercicio5a.c.

4.6.3. Documentación de las funciones

4.6.3.1. int main (void)

main del Ejercicio5a

Programa en el que cada proceso tiene solo 1 hijo

Definición en la línea 19 del archivo ejercicio5a.c.

4.7. Referencia del Archivo ejercicio5b.c

Ejercicio5a.

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

Dependencia gráfica adjunta para ejercicio5b.c:

'defines'

- #define [NUM_PROC](#) 6

Funciones

- int [main](#) (void)
main del ejercicio5b

4.7.1. Descripción detallada

Ejercicio5a.

Autor

Lucia Rivas Molina lucia.rivasmolina@estudiante.uam.es

Daniel Santo-Tomas Lopez daniel.santo-tomas@estudiante.uam.es

Fecha

2018/03/07

4.7.2. Documentación de los 'defines'

4.7.2.1. #define NUM_PROC 6

Definición en la línea 13 del archivo ejercicio5b.c.

4.7.3. Documentación de las funciones

4.7.3.1. int main (void)

main del ejercicio5b

Programa en el que el padre tiene 3 hijos

Definición en la línea 19 del archivo ejercicio5b.c.

4.8. Referencia del Archivo ejercicio6.c

Ejercicio6.

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

Dependencia gráfica adjunta para ejercicio6.c:

Funciones

- int [main](#) (void)

main del ejercicio6

4.8.1. Descripción detallada

Ejercicio6.

Autor

Lucia Rivas Molina lucia.rivasmolina@estudiante.uam.es

Daniel Santo-Tomas Lopez daniel.santo-tomas@estudiante.uam.es

Fecha

2018/03/07

4.8.2. Documentación de las funciones

4.8.2.1. `int main (void)`

main del ejercicio6

Programa en el que comprobamos que el padre no puede acceder al contenido del hijo

Definición en la línea 17 del archivo ejercicio6.c.

4.9. Referencia del Archivo ejercicio8.c

Ejercicio8.

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

Dependencia gráfica adjunta para ejercicio8.c:

'defines'

- `#define MAX_TAM 15`
- `#define ERROR -1`
- `#define OK 0`

Funciones

- `int ejecutar (char *tipo, char *programa)`
Función que ejecuta el hijo.
- `int main (int argc, char **argv)`
Funcion main.

4.9.1. Descripción detallada

Ejercicio8.

Autor

Lucía Rivas Molina lucia.rivasmolina@estudiante.uam.es
Daniel Santo-Tomas Lopez daniel.santo-tomas@estudiante.uam.es

Fecha

2018/03/07

4.9.2. Documentación de los 'defines'

4.9.2.1. #define ERROR-1

Definición en la línea 17 del archivo ejercicio8.c.

4.9.2.2. #define MAX_TAM 15

Definición en la línea 16 del archivo ejercicio8.c.

4.9.2.3. #define OK 0

Definición en la línea 18 del archivo ejercicio8.c.

4.9.3. Documentación de las funciones

4.9.3.1. int ejecutar (char * *tipo*, char * *programa*)

Función que ejecuta el hijo.

Parámetros

| | |
|-----------------|--|
| <i>tipo</i> | : tipo de exec que queremos (-l, -lp...) |
| <i>programa</i> | : nombre del programa a ejecutar(ls) |

Devuelve

ERROR en caso de error y OK si se ha ejecutado

Definición en la línea 26 del archivo ejercicio8.c.

4.9.3.2. `int main (int argc, char ** argv)`

Funcion main.

Programa que ejecuta los programas pasados como parámetros

Parámetros

| | |
|------------------|---|
| <i>programas</i> | a ejecutar |
| <i>exec</i> | que queremos ejecutar (-l -lp -v o -vp) |

Definición en la línea 92 del archivo ejercicio8.c.

4.10. Referencia del Archivo ejercicio9.c

Ejercicio9.

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

Dependencia gráfica adjunta para ejercicio9.c:

'defines'

- `#define MAX_TAM 100`

Funciones

- `double factorial (double a)`
Calcula el factorial del numero pasado como parametro.
- `double combinatoria (double a, double b)`
Calcula el coeficiente binomial de a sobre b.
- `double sumAbs (double a, double b)`
Calcula la suma de dos valores absolutos.
- `int main (void)`
Funcion main.

4.10.1. Descripción detallada

Ejercicio9.

Autor

Lucia Rivas Molina lucia.rivasmolina@estudiante.uam.es

Daniel Santo-Tomas Lopez daniel.santo-tomas@estudiante.uam.es

Fecha

2018/03/07

4.10.2. Documentación de los 'defines'

4.10.2.1. #define MAX_TAM 100

Definición en la línea 15 del archivo ejercicio9.c.

4.10.3. Documentación de las funciones

4.10.3.1. double combinatoria (double *a*, double *b*)

Calcula el coeficiente binomial de *a* sobre *b*.

Parámetros

| | |
|---------------|---------------|
| <i>numero</i> | real <i>a</i> |
| <i>numero</i> | real <i>b</i> |

Devuelve

a sobre *b*

Definición en la línea 35 del archivo ejercicio9.c.

4.10.3.2. double factorial (double *a*)

Calcula el factorial del numero pasado como parametro.

Parámetros

| | |
|-----------|-------------|
| <i>un</i> | numero real |
|-----------|-------------|

Devuelve

factorial del numero

Definición en la línea 23 del archivo ejercicio9.c.

4.10.3.3. int main (void)

Funcion main.

Programa que lanza cuatro hijos y cada uno realiza una operación matemática

Definición en la línea 56 del archivo ejercicio9.c.

4.10.3.4. double sumAbs (double *a*, double *b*)

Calcula la suma de dos valores absolutos.

Parámetros

| | |
|----------------|----------|
| <i>primer</i> | operando |
| <i>segundo</i> | operando |

Devuelve

suma de los valores absolutos

Definición en la línea 47 del archivo ejercicio9.c.

Índice alfabético

cadena
 Estructura, 5
combinatoria
 ejercicio9.c, 18
dim
 Matriz, 6
ERROR
 ejercicio8.c, 16
ejecutar
 ejercicio8.c, 16
ejercicio12a.c, 7
ejercicio12b.c, 7
 main, 8
 primerosNPrimos, 8
 TAM_HILO, 8
ejercicio13.c, 8
 MAX_TAM, 9
 main, 10
 multiplica_matriz, 10
 NUM_HILOS, 9
 RETARDO, 10
ejercicio4.c, 10
 main, 11
 NUM_PROC, 11
ejercicio4b.c, 11
 main, 12
 NUM_PROC, 12
ejercicio5a.c, 12
 main, 13
 NUM_PROC, 13
ejercicio5b.c, 13
 main, 14
 NUM_PROC, 14
ejercicio6.c, 14
 main, 15
ejercicio8.c, 15
 ERROR, 16
 ejecutar, 16
 MAX_TAM, 16
 main, 16
 OK, 16
ejercicio9.c, 17
 combinatoria, 18
 factorial, 18
 MAX_TAM, 18
 main, 18
 sumAbs, 18
Estructura, 5
 cadena, 5
 num, 5
factorial
 ejercicio9.c, 18
fila
 Matriz, 6
hilo
 Matriz, 6
m
 Matriz, 6
MAX_TAM
 ejercicio13.c, 9
 ejercicio8.c, 16
 ejercicio9.c, 18
main
 ejercicio12b.c, 8
 ejercicio13.c, 10
 ejercicio4.c, 11
 ejercicio4b.c, 12
 ejercicio5a.c, 13
 ejercicio5b.c, 14
 ejercicio6.c, 15
 ejercicio8.c, 16
 ejercicio9.c, 18
Matriz, 6
 dim, 6
 fila, 6
 hilo, 6
 m, 6
 mult, 6
mult
 Matriz, 6
multiplica_matriz
 ejercicio13.c, 10
NUM_HILOS
 ejercicio13.c, 9
NUM_PROC
 ejercicio4.c, 11
 ejercicio4b.c, 12
 ejercicio5a.c, 13
 ejercicio5b.c, 14
num
 Estructura, 5
OK
 ejercicio8.c, 16

primerosNPrimos
ejercicio12b.c, [8](#)

RETARDO
ejercicio13.c, [10](#)

sumAbs
ejercicio9.c, [18](#)

TAM_HILO
ejercicio12b.c, [8](#)