



C Piscine

C 08

Resumen: Este documento corresponde al enunciado del módulo C 08 de la C Piscine de 42.

Versión: 14

Índice general

I.	Instrucciones	2
II.	Instrucciones sobre la IA	4
III.	Introducción	7
IV.	Ejercicio 00 : ft.h	8
V.	Ejercicio 01 : ft_boolean.h	9
VI.	Ejercicio 02 : ft_abs.h	11
VII.	Ejercicio 03 : ft_point.h	12
VIII.	Ejercicio 04 : ft_strs_to_tab	13
IX.	Ejercicio 05 : ft_show_tab	15
X.	Entrega y evaluación	16

Capítulo I

Instrucciones

- Esta página será la única referencia: no te fíes de los rumores.
- Antes de empezar a hacer los ejercicios deberás registrarte en el proyecto en tu intranet. Una vez lo hayas hecho, obtendrás tu repositorio Git en el cual debes trabajar.
- Los ejercicios han sido ordenados del más sencillo al más complejo. Por lo tanto, debes hacerlos en orden y asegurarte de que el ejercicio es correcto antes de continuar con el siguiente.
- Los ejercicios de Shell se deben ejecutar con `/bin/bash`.
- Los ejercicios de C se deben compilar con `cc` y utilizando las flags `-Wall -Wextra -Werror`.
- Solamente hay que entregar una función `main()` si lo que se pide es un programa. Si se pide una función se puede entregar el `main()` comentado con la batería de tests que hayas hecho.
- Los ejercicios de C se deben escribir de acuerdo a **la Norma**. Puedes encontrarla en la intranet, en el icono de la brújula, FAQ, sección 42, apartado General Pedagogy como **La Norma**.
- Para ayudarte a comprobar si tus ejercicios cumplen **la Norma** puedes ayudarte de **norminette** - un programa que aplica ciertos requisitos de **la Norma** a tu código. Si tienes archivos o funciones adicionales, también deben cumplir con **la Norma**.
- Lee detenidamente los ejemplos. Podrían dar información que no se especifica en el enunciado...
- Asegúrate de que tus directorios y archivos tienen los permisos adecuados.
- Debes respetar **el procedimiento de entrega** para todos tus ejercicios. Solo el trabajo de tu repositorio **Git** será evaluado.
- **No puedes** dejar en tu directorio **ningún** archivo que no se haya indicado de forma explícita en los enunciados de los ejercicios.

- La evaluación de este proyecto consta de dos fases. Primero, las personas con las que compartes piscina se encargarán de evaluar tus ejercicios. A continuación, serán evaluados por un programa que se llama **la Moulinette**.
- Tus funciones no deben terminar de forma inesperada (segfault, bus error, double free) excepto en el caso de comportamientos indefinidos. Si esto sucede, tu proyecto será considerado no funcional y recibirás un 0 durante la evaluación.
- **La Moulinette** es muy estricta a la hora de evaluar y está completamente automatizada. Es imposible discutir con ella sobre tu nota. Por lo tanto, debes mantener el rigor en tu código para evitar cualquier sorpresa.
- **La Moulinette** utiliza **norminette** para comprobar tus ejercicios, aunque este programa no es exhaustivo en la verificación de **la Norma**. Durante las evaluaciones por parte de las personas de tu Piscina se comprobará que **la Norma** es correcta. Si se incumple el proyecto tendrá una puntuación final de **0**, aunque **norminette** no indique ningún error.
- ¿Tienes alguna pregunta? Pregunta a la persona de tu derecha. Si no puede ayudarte, prueba con la persona de tu izquierda.
- ¡Por **Loki**, por **Freyja**! ¡¡Piensa fuerte!!

Capítulo II

Instrucciones sobre la IA

Contexto

La Piscina de C es intensa. Es tu primer gran desafío en 42: una inmersión profunda en la resolución de problemas, la autonomía y la comunidad.

Durante esta etapa, tu objetivo principal es construir unas bases sólidas, a través del esfuerzo, la repetición y, sobre todo, mediante el **aprendizaje entre pares**.

En la era de la IA, los atajos son fáciles de encontrar. Sin embargo, es importante considerar si el uso que haces de la IA te está ayudando realmente a crecer, o si simplemente te está impidiendo desarrollar habilidades reales.

La Piscina también es una experiencia humana y, por ahora, nada puede reemplazar eso. Ni siquiera la IA.

Para obtener una visión más completa de nuestra postura sobre la IA como herramienta de aprendizaje, como parte del plan de estudios de TIC (Tecnologías de la Información y la Comunicación) y como una expectativa creciente en el mercado laboral, consulta las preguntas frecuentes *FAQ* disponibles en la intranet.

● Mensaje principal

- 👉 Construir fundamentos sólidos sin atajos.
- 👉 Desarrollar de forma real habilidades técnicas y transversales.
- 👉 Experimentar el aprendizaje entre pares de forma real, empezar a aprender a aprender y a resolver nuevos problemas.
- 👉 El proceso de aprendizaje es más importante que el resultado.
- 👉 Aprender sobre los riesgos asociados a la IA y desarrollar prácticas de control efectivas y medidas de mitigación para evitar errores comunes.

● Reglas para la piscina:

- Aplica la lógica y el razonamiento a las tareas asignadas, especialmente antes de recurrir a la IA.
- No deberías pedir respuestas directas a la IA.
- Aprende sobre el enfoque global de 42 sobre la IA.

● Resultados de esta etapa:

Durante esta etapa fundamental, obtendrás los siguientes resultados:

- Obtener Una base sólida técnica y en programación.
- Comprender por qué y cómo la IA puede ser peligrosa durante esta fase.

● Comentarios y ejemplos:

- Si, sabemos que la IA existe. Y sí, puede resolver tus proyectos. Pero estás aquí para aprender, no para demostrar que la IA ha aprendido. No pierdas tu tiempo (o el nuestro) solo para demostrar que la IA puede resolver el problema dado. y que puede resolver tus proyectos, pero estás aquí para aprender. Te recomendamos que inviertas 26 días de aprendizaje de calidad para resolver los retos que hemos pensado para ti.
- Aprender en 42 no va sólo de saber la respuesta. Se trata de desarrollar la capacidad de encontrar una. La IA te da la respuesta directamente, pero eso te impide construir tu propio razonamiento. Y el razonamiento requiere tiempo, esfuerzo y conlleva fracasos. El camino hacia el éxito no debe ser fácil. no tiene nada que ver con saber la respuesta. En 42 aprendemos desarrollando la capacidad para encontrarla. La IA te dará la respuesta directa, lo que impide que desarrolles tu propio razonamiento. Razonar requiere tiempo, esfuerzo y equivocarse muchas veces.
- Piensa que durante los exámenes, la IA no está disponible sin internet, sin smartphones, etc. Te darás cuenta rápidamente si has confiado demasiado en la IA durante tu proceso de aprendizaje. Recuerda que durante los exámenes no tendrás acceso a la IA ni a internet. Te vas a enfrentar a una hoja en blanco donde vas a tener que escribir tu propio código.
- El aprendizaje entre pares te expone a diferentes ideas y enfoques, mejorando tus habilidades interpersonales y tu capacidad de pensar de forma diferente. Eso es mucho más valioso que simplemente chatear con un bot. Así que, ¡que no te supere la timidez!: ¡habla, haz preguntas y aprende con el resto de la piscina!

- Sí, la IA formará parte del plan de estudios, tanto como herramienta de aprendizaje como tema en sí mismo. Incluso tendrás la oportunidad de crear tu propio software de IA. Para aprender más sobre nuestro enfoque progresivo, puedes consultar la documentación disponible en la intranet.

✓ Buenas prácticas:

Me atasco en un nuevo concepto. Le pregunto a alguien cercano cómo lo ha abordado. Hablamos durante 10 minutos y, de repente, todo encaja. Lo entiendo. No entiendo algo concreto del proyecto y no sé cómo continuar. Le pregunto a otra persona de la piscina cómo lo ha abordado, hablamos sobre el tema y, si es necesario, incluso utilizamos otros métodos (papel y boli, dibujos, metáforas, etc.) hasta conseguir entenderlo.

✗ Mala práctica:

Utilizo la IA en secreto, copio un código que parece correcto. Durante la evaluación entre pares, no puedo explicar nada. Suspenso. Durante el examen, sin IA, me vuelvo a atascar. Suspenso.

Capítulo III

Introducción

Esto es lo que nos cuenta la enciclopedia colaborativa *Wikipedia* sobre el ornitorrinco:

El ornitorrinco (*Ornithorhynchus anatinus*) es una especie de pequeño mamífero semiacuático endémico del este de Australia, incluida Tasmania. Es una de las cinco especies del orden de los monotremas, único orden de mamíferos que ponen huevos en lugar de parir crías completamente formadas (las otras cuatro especies son equidnas). Es la única especie superviviente de la familia Ornithorhynchidae y del género Ornithorhynchus, aunque se hayan descubierto una gran cantidad de fragmentos de especies fósiles de esta familia y de este género.


La apariencia extraña de este mamífero ponedor de huevos, dotado de espolones venenosos, con la mandíbula córnea parecida al pico de un pato, una cola que recuerda a un castor, que le sirve tanto de timón en el agua como de reserva de grasa, y patas de nutria sorprendió enormemente a los primeros exploradores que lo descubrieron; buena parte de los naturalistas europeos pensaron que se trataba de una broma. Es uno de pocos mamíferos venenosos: el macho lleva sobre las patas traseras un espolón que puede liberar veneno capaz de provocar un dolor intenso a un humano. Las características singulares del ornitorrinco lo convierten en un sujeto de estudio importante para entender mejor la evolución de las especies animales y se ha convertido en uno de los símbolos de Australia: ha sido utilizado como mascota en acontecimientos nacionales y aparece en el dorso de la moneda australiana de 20 céntimos.

Fue cazado hasta principios del siglo XX por su piel, pero actualmente está protegido. A pesar de que los programas de reproducción en cautividad han tenido un éxito bastante limitado y de que sea un animal vulnerable a los efectos de la contaminación, todavía no se considera que la especie esté en peligro.

Este enunciado no tiene nada que ver con el ornitorrinco.

Capítulo IV

Ejercicio 00 : ft.h


	Ejercicio: 00
ft.h	
Directorio de entrega: <i>ex00/</i>	
Archivos a entregar: ft.h	
Funciones autorizadas: Ninguna	

- Crea tu archivo **ft.h**
- Debe contener todos los prototipos de las siguientes funciones:

```
void    ft_putchar(char c);
void    ft_swap(int *a, int *b);
void    ft_putstr(char *str);
int     ft_strlen(char *str);
int     ft_strncmp(char *s1, char *s2);
```

Capítulo V

Ejercicio 01 : ft_boolean.h

	Ejercicio: 01
	ft_boolean.h
	Directorio de entrega: <i>ex01/</i>
	Archivos a entregar: ft_boolean.h
	Funciones autorizadas: Ninguna

- Escribe un archivo `ft_boolean.h` que haga que compile y funcione correctamente el main siguiente:

```
#include "ft_boolean.h"

void      ft_putstr(char *str)
{
    while (*str)
        write(1, str++, 1);
}

t_bool     ft_is_even(int nbr)
{
    return ((EVEN(nbr)) ? TRUE : FALSE);
}

int        main(int argc, char **argv)
{
    (void)argv;
    if (ft_is_even(argc - 1) == TRUE)
        ft_putstr(EVEN_MSG);
    else
        ft_putstr(ODD_MSG);
    return (SUCCESS);
}
```

- Este programa mostrará:

```
I have an even number of arguments.
```

- O:

```
I have an odd number of arguments.
```


- seguido de un salto de línea, en caso necesario.



Para este ejercicio, la Norminette debe ser ejecutada con el flag `-R CheckDefine`. La Moulinette también lo utilizará.

Capítulo VI

Ejercicio 02 : ft_abs.h

	Ejercicio: 02
	ft_abs.h
	Directorio de entrega: <i>ex02/</i>
	Archivos a entregar: ft_abs.h
	Funciones autorizadas: Ninguna

- Escriba una macro ABS que sustituya su parámetro por su valor absoluto:


```
#define ABS(Value)
```



Para este ejercicio, la Norminette debe ser ejecutada con el flag *-R CheckDefine*. La Moulinette también lo utilizará.

Capítulo VII

Ejercicio 03 : ft_point.h

	Ejercicio: 03
	ft_point.h
	Directorio de entrega: <i>ex03/</i>
	Archivos a entregar: ft_point.h
	Funciones autorizadas: Ninguna

- Escriba un archivo **ft_point.h** que haga que compile el main siguiente:

```
#include "ft_point.h"


void      set_point(t_point *point)
{
    point->x = 42;
    point->y = 21;
}

int       main(void)
{
    t_point      point;

    set_point(&point);
    return (0);
}
```

Capítulo VIII

Ejercicio 04 : ft_strs_to_tab

	Ejercicio: 04
	ft_strs_to_tab
	Directorio de entrega: <i>ex04/</i>
	Archivos a entregar: ft_strs_to_tab.c
	Funciones autorizadas: malloc, free

- Crea una función que reciba como parámetro un array de cadenas de caracteres y su tamaño.
- El prototipo de la función deberá ser el siguiente:

```
struct s_stock_str *ft_strs_to_tab(int ac, char **av);
```

- La función debe transformar cada elemento del array de cadenas de caracteres en estructura.
- La estructura será definida en el archivo **ft_stock_str.h** de la siguiente forma:


```
typedef struct      s_stock_str
{
    int  size;
    char *str;
    char *copy;
}        t_stock_str;
```

- **size** es el tamaño de la cadena de caracteres;
 - **str** es la cadena de caracteres;
 - **copy** es una copia de la cadena de caracteres;
- Debes conservar el orden de los elementos de **av**.

- Se tendrá que asignar memoria a la tabla de estructuras y el último elemento tendrá 0 como valor de `str`, para indicar el final de la tabla.
- Si surge un error de asignación de memoria la función deberá retornar un puntero `NULL`.
- Se probará tu función con el `ft_show_tab` (ejercicio siguiente). ¡Haz lo que sea preciso para que funcione!

Capítulo IX

Ejercicio 05 : ft_show_tab

	Ejercicio: 05
	ft_show_tab
	Directorio de entrega: <i>ex05/</i>
	Archivos a entregar: ft_show_tab.c
	Funciones autorizadas: write

- Crea una función que muestre el contenido del array creado con la función anterior.
- El prototipo de la función deberá ser el siguiente:

```
void ft_show_tab(struct s_stock_str *par);
```

- La estructura es la misma que la del ejercicio anterior y se definirá en el archivo **ft_stock_str.h**.
- Para cada elemento de la tabla se mostrará:
 - la cadena de caracteres irá seguida de un salto de línea
 - el tamaño irá seguido de un salto de línea
 - la copia de la cadena de caracteres (que podrá haber sido modificada) irá seguida de un salto de línea
- Se probará tu función con el **ft_strs_to_tab** (ejercicio anterior). ¡Haz lo que sea preciso para que funcione!

Capítulo X

Entrega y evaluación

Entrega tu proyecto en tu repositorio `Git` como de costumbre. Solo el trabajo entregado en el repositorio será evaluado durante la defensa. No dudes en comprobar varias veces los nombres de los archivos para verificar que sean correctos.



Sólo necesitas entregar los archivos requeridos por el enunciado de este proyecto.