

Set

Entrega y prueba de evaluación.....	1
Material que se debe entregar	2
Calificación del proyecto	2
Descripción del juego <i>Set</i>	2
Nuestra versión de <i>Set</i>	5
Normas de implementación.....	6
Tipos abstractos de datos	6
Librería entorno	6
TAD Carta	7
TAD Baraja.....	7
TAD Tablero.....	7
TAD Juego	8
Planificación	8
Documentación	9
Documentación interna	9
Documentación externa	9

A continuación, se detallan los requisitos para la elaboración del proyecto de programación obligatorio de la asignatura Introducción a la programación para el curso 2021/2022.

El objetivo de este proyecto es diseñar, implementar en C++ y documentar una versión del juego *Set* con ayuda de una serie de módulos ya escritos para manejar el entorno gráfico y la interacción.

¡IMPORTANTE!

Se deben cumplir todos los requisitos que se indican en este documento.

Este proyecto puede realizarse **en grupos de dos o individualmente**. La cantidad de trabajo que hay que realizar está pensada para dos estudiantes, pero, opcionalmente, se puede hacer de manera individual.

En el apartado correspondiente del aula virtual de la asignatura se puede encontrar toda la información sobre el proyecto, un ejecutable de prueba como modelo de lo que debe entregarse, un proyecto base con los archivos del entorno, una plantilla para la documentación y la rúbrica de evaluación. Además, se puede usar el foro de la asignatura para preguntar dudas y aclaraciones.

Es responsabilidad de cada estudiante la custodia y protección de su proyecto. **Solo las personas del grupo y el profesorado de la asignatura pueden ver el código desarrollado para un proyecto. Si ves el programa de otra persona, estás copiando. Si dejas tu proyecto a otra persona, estás copiando.**

Se utilizará un software de detección de copias en los programas entregados. En caso de encontrar similitudes en **partes significativas** de los programas, **todas las personas implicadas** tendrán una nota de SUSPENSO (0) en la asignatura.

ENTREGA Y PRUEBA DE EVALUACIÓN

La entrega final del proyecto de programación se realizará mediante la actividad correspondiente del aula virtual.

El plazo de presentación se avisará con suficiente antelación. En cada convocatoria será siempre un día antes del día del examen oficial de la asignatura.

Cada miembro del grupo deberá entregar exactamente la misma información. Tanto en la documentación interna como en la externa deberán aparecer los nombres de las personas que han realizado el trabajo.

El día del examen final de la asignatura se realizarán las correspondientes pruebas de evaluación del proyecto en el horario que se publicará en la convocatoria oficial. Esta prueba solo hay que hacerla si se ha entregado el proyecto en el plazo indicado.

La prueba se realizará de manera individual, independientemente de que el proyecto se haya realizado en pareja o individualmente.

Es necesario comprobar que se ha enviado la información correctamente y que está accesible en el aula virtual.

MATERIAL QUE SE DEBE ENTREGAR

Al entregar el proyecto se debe presentar un único fichero comprimido (.zip o .tar.gz) con el nombre de las personas que han hecho el programa, **nombre1_nombre2.zip** (o .tar.gz) que contenga los siguientes archivos:

- La documentación externa del proyecto en formato ODT, DOC o PDF.
- El archivo comprimido (.zip o .tar.gz) con la exportación del proyecto, con todos los archivos necesarios para poder compilarlo y ejecutarlo.
- Otros proyectos comprimidos con las ampliaciones (si se hacen).

Nombre1 representa el nombre completo con los dos apellidos del primer componente del grupo (por orden alfabético) y nombre2 al del segundo. Por ejemplo, si el proyecto lo realizaran dos profesoras de la asignatura el fichero debería llamarse MariscalAraujoMAngelos_VicenteChicoteCristina.zip (o .tar.gz).

CALIFICACIÓN DEL PROYECTO

Un requisito previo a la evaluación del proyecto es la superación de la **prueba de evaluación**.

La prueba de evaluación del proyecto consistirá en unas preguntas sobre el proyecto y una modificación del funcionamiento básico. Las respuestas y la modificación se harán en papel, aunque se podrá consultar el código del proyecto en el ordenador.

Si no se supera la prueba de evaluación, la nota será 2.

Si es copia de otro proyecto, la nota será 0 en el bloque y en toda la asignatura para todas las personas implicadas.

La calificación se hará en función del programa, de la documentación y de la prueba de evaluación del proyecto.

En los proyectos realizados de forma individual, la nota del bloque de proyecto se corresponderá con la nota de la evaluación del proyecto.

En los proyectos realizados en pareja, si ambos miembros del grupo superan la prueba de evaluación en la misma convocatoria y la calificación del proyecto es superior a 5, la **nota final del bloque del proyecto** se aumentará en 1 punto. En cualquier otro caso, se restará 1 punto a la calificación obtenida en el bloque del proyecto en la convocatoria en la que superen la prueba de evaluación.

DESCRIPCIÓN DEL JUEGO SET

Set es un juego de mesa para uno o más jugadores. Se pueden encontrar muchas versiones distintas para jugar en un navegador o en el móvil.

Por ejemplo, https://smart-games.org/en/set/submit_set.

Para teléfonos Android se puede encontrar el juego *Find a set!* en Google Play.

El objetivo del juego es identificar conjuntos (*set*) de 3 cartas a partir de las cartas que se van mostrando.

Cada carta tiene 4 características distintas:

- Número: en cada carta hay 1, 2 o 3 elementos, todos ellos iguales.



- Forma: los elementos de cada carta pueden ser elipses, rombos u ondas. Todos los elementos de cada carta tienen la misma forma.



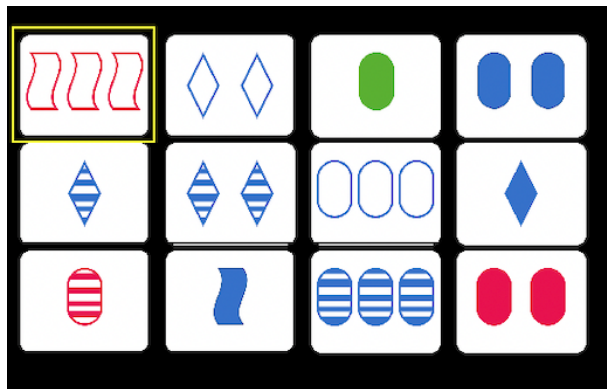
- Color: los elementos de cada carta pueden ser azules, verdes o rojos. Todos los elementos de cada carta tienen el mismo color.



- Relleno: los elementos de cada carta pueden estar vacíos, rayados o rellenos. Todos los elementos de cada carta tienen el mismo relleno.



Se parte de una baraja de 81 cartas distintas (las 81 combinaciones diferentes que se pueden formar con las 3 posibilidades de cada una de las 4 características: $3^4 = 81$). Inicialmente, en el tablero se suelen mostrar 12 cartas.

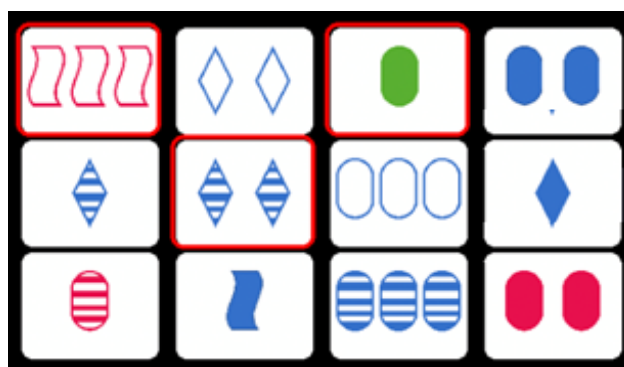


Con las teclas de los cursores ($\leftarrow \uparrow \downarrow \rightarrow$) podremos ir desplazándonos por las cartas del tablero. Para seleccionar una carta deberemos pulsar la tecla *Enter*. Una vez seleccionadas 3 cartas, si forman un conjunto (*set*), esas cartas desaparecerán, se sustituirán por otras de la baraja y se incrementará el número de puntos conseguidos.

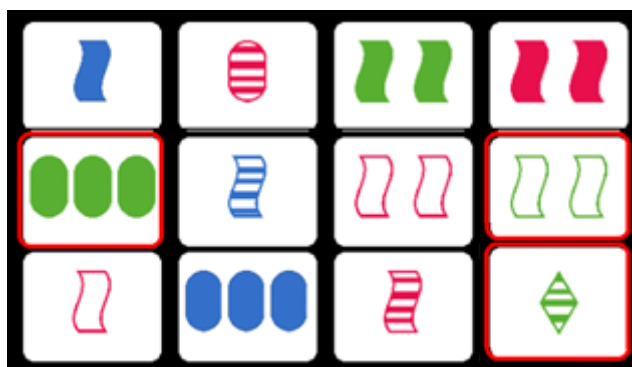
En la versión original, el juego termina cuando no se puede formar ningún *set*.

Un conjunto o *set* estará formado por 3 cartas cuyas cuatro características sean iguales o distintas en todas ellas.

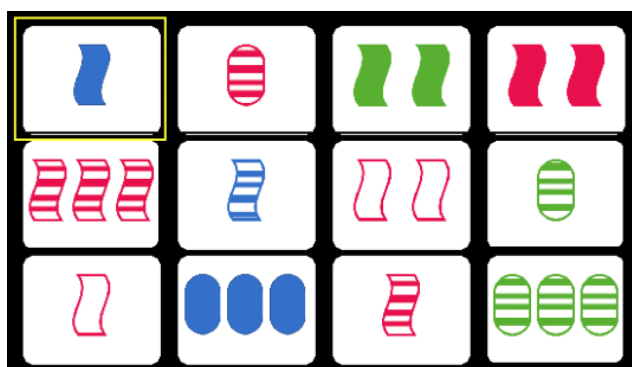
Por ejemplo, en la imagen siguiente, las tres cartas seleccionadas (enmarcadas en rojo) forman un *set* ya que todas ellas tienen distinto número, distinta forma, distinto color y distinto relleno.



La imagen siguiente muestra otro ejemplo de 3 cartas seleccionadas que también forman un *set* ya que todas tienen el mismo color y todas tienen distinto número, forma y relleno.

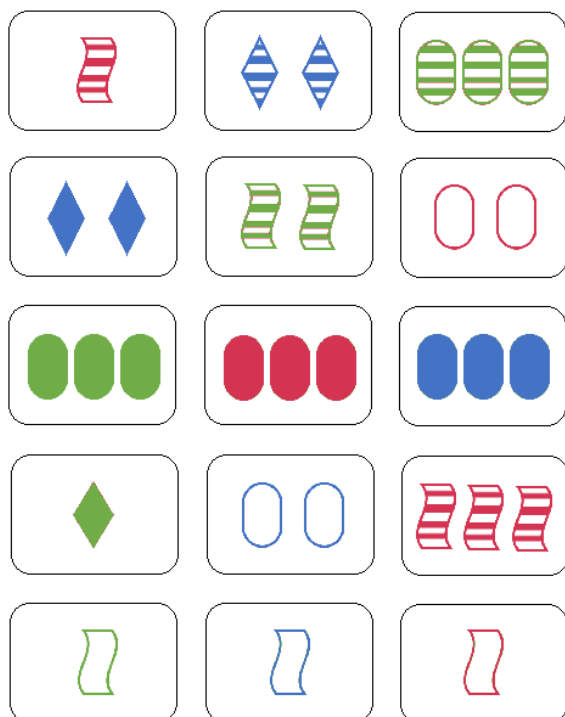


Cada vez que se encuentra un *set*, el programa eliminará las 3 cartas seleccionadas en la jugada anterior y, en su lugar, mostrará 3 cartas nuevas. Las restantes cartas permanecerán en su sitio. La siguiente imagen muestra una posible configuración del tablero tras seleccionarse las 3 cartas de la imagen anterior.

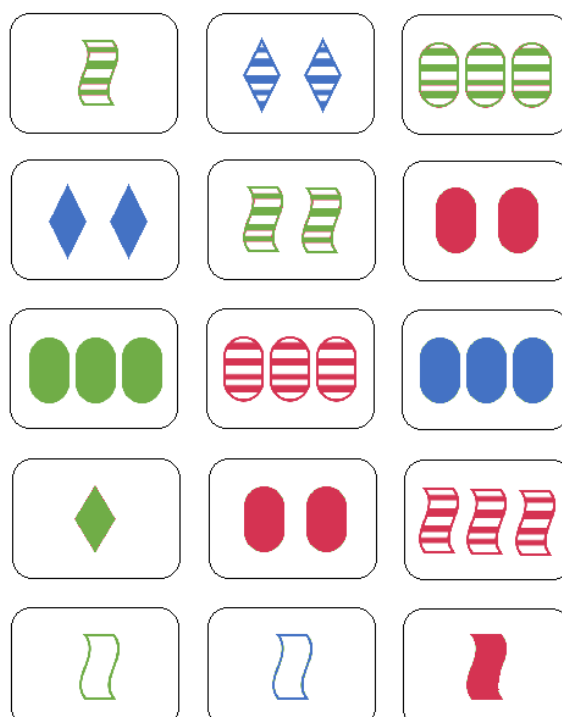


A continuación, se muestran algunos ejemplos más de tríos de cartas que forman *set* y tríos de cartas que no.

Sets correctos



Sets incorrectos



NUESTRA VERSIÓN DE SET

Versión básica

El programa servirá para una única partida.

Hay un fichero de configuración donde se determinarán las características iniciales de la partida. Su utilización se explica en el apartado correspondiente a la librería *entorno*.

La configuración inicial se cargará con la información que haya en el fichero de configuración.

Se podrá configurar el número de columnas que ocuparán las cartas iniciales. Podrá haber entre 3 y 5 columnas de 3 cartas cada una. Por tanto, podrá haber 9, 12 o 15 cartas inicialmente.

La persona que juegue utilizará las teclas de los cursores del teclado (←↑↓→) para mover la marca amarilla que indica la carta actual.

Al pulsar la tecla **Enter** se seleccionará la carta actual con un recuadro rojo. Si la carta ya estaba seleccionada, dejará de estarlo y se eliminará el recuadro rojo.

Cuando haya 3 cartas seleccionadas, se comprobará si forman o no un *set*.

- Si no forman un *set*, se eliminarán las marcas de selección de las tres cartas y se continuará jugando.
- Si forman un *set*, se borrarán esas 3 cartas, se sustituirán por otras 3 cartas que no hayan aparecido todavía y se continuará jugando.

El juego termina cuando se haya alcanzado la puntuación indicada en el fichero de configuración o cuando se pulse la tecla *Esc*.

Cada vez que se forme un *set* se incrementará la puntuación con un valor fijado en el fichero de configuración.

La nota máxima que se puede obtener en el proyecto con esta versión básica es de 8.

Ampliaciones

Existe la posibilidad de presentar, junto con el proyecto básico explicado anteriormente, una versión ampliada.

Con estas ampliaciones se podrá llegar a obtener la calificación de 10 (personas que presenten individualmente el proyecto) o 12 (parejas), **siempre que el proyecto básico esté correctamente implementado y documentado.**

Si existen errores graves en la versión básica o en la documentación no se tendrán en cuenta las ampliaciones.

Ampliaciones propuestas para todos los trabajos (individuales o en pareja):

Se puede incluir **solo una de las dos** propuestas de este bloque.

- **Columna:** al pulsar la tecla F1 se añade una nueva columna de cartas a la derecha. Se restará la cantidad de puntos indicada en el fichero de configuración. No se podrá añadir la columna si no se ha alcanzado esa puntuación. Solo se puede añadir una nueva columna una vez. (Hasta 1 punto.)
- **Ayuda:** al pulsar la tecla F2 se resaltarán durante 2 segundos tres cartas que formen un conjunto. Se restará la cantidad de puntos indicada en el fichero de configuración. No se podrá obtener la ayuda si no se ha alcanzado esa puntuación. (Hasta 2 puntos.)

Ampliaciones propuestas solo para trabajos realizados en pareja:

Se pueden incluir las dos ampliaciones propuestas para todos los trabajos y la siguiente ampliación. Por tanto, se pueden conseguir 4 puntos como máximo.

- **Resolución automática:** el juego se irá resolviendo de forma automática sin intervención de ninguna persona. En esta ampliación, el juego terminará cuando no se pueda formar ningún *set*. (Hasta 1 punto.)

No se admitirá ninguna ampliación distinta a las propuestas.

Las ampliaciones tienen que estar en un proyecto diferente al de la versión básica del juego. También debe incluirse en la documentación externa un pequeño resumen de lo que se ha cambiado con respecto a la versión básica.

¡Atención! Algunas de estas ampliaciones pueden requerir una inversión considerable de tiempo por lo que cada estudiante debe sopesar si abordarlas o no, considerando su carga de trabajo en ésta y otras asignaturas.

NORMAS DE IMPLEMENTACIÓN

El proyecto debe implementarse usando exclusivamente programación imperativa con C++, usando las estructuras de datos y librerías que se han visto en clase.

El proyecto debe ejecutarse correctamente en la máquina virtual utilizada en la asignatura y en los ordenadores instalados en los laboratorios.

Se debe usar la librería *entorno* entregada por los profesores para gestionar el entorno gráfico del juego. **Esta librería no puede modificarse.** Para que funcione, es necesario tener instalada la librería *allegro5* (en la máquina virtual y en los laboratorios ya está instalada).

Se deben definir, como mínimo, TAD para representar una carta, la baraja, el tablero y el juego.

Cada TAD o librería estará definido por un fichero .h y otro .cpp.

Cada TAD o librería deberá ir acompañado de su correspondiente **librería de pruebas** (definidas, a su vez, por un fichero .h y otro cpp). No es necesario implementar pruebas para la librería *entorno* ni para el *TAD Juego*.

En el aula virtual de la asignatura hay un proyecto base de *Eclipse (JuegoSet.zip)*, que debe utilizarse como punto de partida para el trabajo. Este proyecto incluye la librería *entorno* (*entorno.h* y *entorno.cpp*), varios ejemplos de ficheros de configuración y un pequeño programa de ejemplo. El proyecto base está configurado ya con la información sobre las librerías necesarias.

En el aula virtual también hay un programa ejecutable que muestra el aspecto y el funcionamiento básico del juego.

TIPOS ABSTRACTOS DE DATOS

Tras un análisis del problema, surge la necesidad de modelar los siguientes elementos que se manejan con las librerías o tipos abstractos correspondientes:

- La **librería entorno** es la que se encargará de gestionar la interfaz gráfica de la aplicación y la interacción con la persona que juegue.
- El **TAD Carta** definirá la estructura de datos necesaria para almacenar la información de una carta y ofrecerá las operaciones necesarias para su inicialización, consulta, etc.
- El **TAD Baraja** definirá la estructura de datos necesaria para almacenar la información de la baraja de cartas y las operaciones necesarias para inicializarla y gestionarla (generar las cartas, sacar una nueva carta, etc.).
- El **TAD Tablero** definirá la estructura de datos necesaria para almacenar el conjunto de cartas visibles en el juego y ofrecerá operaciones para inicializarlo, añadir y eliminar cartas, comprobar si las cartas seleccionadas forman un *set*, etc.
- El **TAD Juego** definirá la estructura de datos necesaria para almacenar la información sobre la partida y será el responsable de iniciar el juego a partir de la información que haya en el fichero de configuración y de controlar el juego, encargándose de gestionar las teclas que se pulsan, la puntuación, los movimientos del cursor y el fin del juego, así como de mantener la coherencia entre la información que se almacena en el tablero y en la baraja y lo que se representa en pantalla en cada instante del juego.

LIBRERÍA ENTORNO

Con el fin de facilitar el desarrollo del proyecto se entrega (dentro del proyecto base disponible en el aula virtual), una librería con los módulos necesarios para gestionar el entorno gráfico del juego. **Esta librería no debe modificarse.** Desde los módulos desarrollados solo se deben usar las operaciones definidas en el fichero *entorno.h*.

(Si alguna persona tiene dificultad para distinguir los colores propuestos, que contacte con el profesorado de la asignatura para que le pasemos una versión modificada del entorno.)

En **entorno.h** aparecen las constantes definidas y las funciones de manejo del entorno (con su especificación con pre/postcondiciones). Podemos suponer que todas las funciones del entorno tienen un coste constante, $O(1)$.

También se define el tipo enumerado con los valores que devuelve la función **entornoLeerTecla**. Este enumerado incluye únicamente las teclas aceptadas en juego (cursores, Enter, F1, F2 y Escape). Si se pulsa cualquier otra tecla, la función **entornoLeerTecla** devolverá el valor *TNada*.

El juego se configurará a partir de la información del fichero **set.cnf** que debe estar ubicado en el directorio del proyecto.

La estructura de este fichero de texto, que nos permite cargar la configuración inicial del juego, es la siguiente:

- En la primera línea, el número de columnas iniciales, **n**. Este valor puede ser 3, 4 o 5.
- En la segunda línea, el número de puntos que se consiguen al formar un **set (pc)**, que será un entero positivo.
- En la tercera línea, el número de puntos que hay que alcanzar para terminar el juego (**p**), que será un entero positivo.
- En la cuarta línea, el número de puntos que hay que tener para poder usar la ayuda o añadir una columna (**pa**), que será un entero no negativo (puede ser 0). Cada vez que se use la ayuda o se añada una columna se descontarán **pa** puntos del marcador.

Se puede suponer que el fichero de configuración es correcto. Si es incorrecto, el comportamiento de la aplicación no tiene que ser correcto.

Hay una operación en la librería entorno que devuelve toda la información que hay en el fichero de configuración.

TAD CARTA

El TAD Carta gestionará la información de una carta.

La estructura de datos para almacenar una carta debe guardar el valor de cada una de las 4 características de la carta (número, forma, color y relleno).

Entre otras, se pueden necesitar las siguientes operaciones:

- Iniciar una carta con sus cuatro características
- Obtener el valor de cada una de las características

TAD BARAJA

El TAD Baraja gestionará la información de las 81 cartas distintas que puede haber en el juego.

La estructura de datos debe poder almacenar el conjunto de cartas que todavía no se han utilizado.

Entre otras, se pueden necesitar las siguientes operaciones:

- Crear la baraja inicial de 81 cartas distintas
- Sacar una carta de la baraja
- Devolver el número de cartas que quedan en la baraja

Cuando se vayan sacando las cartas de la baraja se debe garantizar que vayan saliendo en orden aleatorio. Esto puede hacerse de muchas formas distintas.

TAD TABLERO

El TAD Tablero gestiona la información de las cartas disponibles para jugar en cada momento. **Este TAD no gestiona las teclas que se pulsan durante el juego ni muestra nada por pantalla** (de eso se encarga el TAD Juego).

La estructura de datos debe guardar:

- Espacio suficiente para almacenar el máximo número de cartas que se pueden mostrar
- El número de columnas reales que se están mostrando en un instante determinado

Algunas de las operaciones que pueden ser útiles son las siguientes (pueden ser necesarias algunas más, o se pueden agrupar o dividir en otras similares; no siempre se indican todos los parámetros que deben pasarse):

- Iniciar el tablero con los datos necesarios
 - Devolver el número de columnas que hay
 - Devolver el número de cartas que quedan en la baraja
-

-
- Obtener una carta concreta del tablero
 - Obtener cada una de las características de una carta concreta del tablero
 - Seleccionar o dejar de seleccionar una carta concreta del tablero
 - Obtener si está o no seleccionada una carta concreta del tablero
 - Añadir una carta en una posición concreta del tablero
 - Comprobar si las cartas seleccionadas en el tablero forman un *set*

TAD JUEGO

El TAD Juego es el encargado de la gestión del juego completo. Será el que gestione el tablero, la puntuación, la interacción con el usuario a través del teclado y la actualización del entorno gráfico del juego (la pantalla).

La estructura de datos correspondiente al juego debe guardar, como mínimo:

- El tablero
- La baraja de cartas que quedan
- La puntuación de la partida

Como mínimo, deberá incluir tres módulos:

- Un módulo **iniciar** que inicie la estructura de datos del juego, según la configuración del fichero *set.cnf*
- Un módulo **jugar** que realice la gestión general del juego (gestionar las teclas que se pulsen, actualizar el tablero y la pantalla, dar el juego por finalizado, etc.)
- Un módulo **terminar** que termine el juego, mostrando un mensaje de despedida y cerrando el entorno gráfico.

PLANIFICACIÓN

El trabajo de implementación y documentación de este proyecto está planificado para realizarse en menos de 50 horas entre dos estudiantes que hayan realizado un seguimiento correcto de las clases, sesiones de laboratorio y actividades propuestas hasta el momento.

A continuación, se presenta una posible planificación, con las horas de dedicación y las principales tareas que hay que desarrollar. Las tareas están detalladas por horas de trabajo individual; algunas serán conjuntas y otras por separado. Por ejemplo, la definición del tablero podría realizarse en una reunión de una hora de duración.

Hay que ir anotando los tiempos dedicados a cada una de las tareas para incluirlos posteriormente en la documentación externa del proyecto.

Horas	Tarea
4	Lectura de la documentación inicial, planificación del trabajo en grupo (2 horas cada persona)
2	Prueba del proyecto base
2	Diseño general de la aplicación y de los TAD necesarios
4	Diseño e implementación del TAD Carta (estructura de datos, implementación de operaciones y pruebas)
4	Diseño e implementación del TAD Baraja (estructura de datos, implementación de operaciones y pruebas)
2	Definición del TAD tablero (estructura de datos y operaciones necesarias)
8	Implementación del TAD tablero (juegos de pruebas y operaciones)
10	Definición e implementación del TAD juego (estructura de datos y operaciones)
4	Pruebas de integración del proyecto

6	Ampliaciones del proyecto
4	Escritura final de la documentación (una parte de esta tarea se debe realizar a lo largo de todo el proceso, mientras se implementan los TAD)

Total: 50 horas

DOCUMENTACIÓN

El proyecto debe ir acompañado de su correspondiente documentación interna y externa.

DOCUMENTACIÓN INTERNA

En el fichero .h de cada TAD se debe incluir la especificación con pre/postcondiciones y la complejidad de cada operación.

En el fichero .h de las librerías de prueba de cada TAD se deben incluir, con comentarios, el diseño de las pruebas que se hacen en cada módulo.

DOCUMENTACIÓN EXTERNA

En el aula virtual se puede encontrar una plantilla que sirve de guía para la redacción de la documentación externa, con la información que debe contener.

El formato de la página de la cubierta debe ser el indicado en la plantilla de la documentación, que incluye la identificación de los estudiantes, el grupo al que pertenecen y su profesor de laboratorio.

A continuación, se explican brevemente los distintos apartados que, como mínimo, debe contener la documentación externa entregada.

Introducción

Toda documentación de programas debe incluir una breve introducción sobre el software desarrollado, explicando el objetivo principal y los requisitos satisfechos.

Análisis y diseño

Una vez que conocemos el problema (descrito en la introducción), debemos analizar qué entidades intervienen en el mismo, qué estructuras de datos son las más adecuadas para representarlos y qué operaciones pueden realizar.

La documentación relativa al análisis facilita la comprensión del problema ya que, sin entrar en demasiados detalles, se describen los TAD que posteriormente serán implementados y las principales decisiones tomadas.

En el diseño de la aplicación se detalla cada uno de los TAD utilizados para desarrollar la aplicación, explicando tanto su estructura como sus operaciones.

Esta sección debe comenzar con el esquema de los TAD utilizados, pasando después a describir cada uno de ellos y el programa principal.

Planificación y tareas

En este apartado de la documentación se debe incluir la planificación inicial del proyecto, las principales tareas realizadas en el desarrollo del programa, quién las ha realizado y el tiempo empleado en cada una de ellas.

Es fundamental incluir, entre las tareas, las reuniones mantenidas, el tiempo empleado en cada una de ellas y los principales acuerdos alcanzados.

Conclusiones y principales problemas

En este apartado de la documentación se deben incluir las principales conclusiones extraídas por los autores del trabajo. Además, se debe reflexionar sobre los problemas encontrados a la hora de desarrollar la aplicación y lo que se ha aprendido.

Los listados con el código no deben incluirse en la documentación externa.
