



Carrera: Técnico Universitario en Programación
Materia: Programación II
Tema: Datos y consignas para resolver el parcial 1

Recuperatorio Parcial 1 - Programación II

Integrantes de la cátedra

Kloster, Daniel - Profesor

Wenner Maximiliano - Ayudante

Contenido

El parcial consta de 4 preguntas que deben resolverse escribiendo directamente en el espacio que el cuestionario dispone, o, para el caso de las dos primeras, haciendo el desarrollo en CodeBlock, copiarlo y luego pegarlo en el cuadro de texto del cuestionario.

Las dos primeras preguntas requieren del desarrollo de funciones, y pueden necesitar del armado de clases; las dos últimas se responden con un texto corto.

Las preguntas para cada alumno se obtienen ingresando en el cuestionario **Recuperatorio Parcial 1 27/06/2022** del aula virtual.

Las preguntas 1 y 2 tienen un puntaje de 3 puntos; la 3 y 4 valen 2 puntos cada una.

El tiempo efectivo de trabajo, una vez aclaradas las dudas sobre el funcionamiento, es de **2 horas**. Pasado ese tiempo habrá un tiempo de 5 minutos para la entrega. **Pasado ese tiempo no puede entregarse el examen, por lo que se considerará como ausente.**

La entrega del parcial debe hacerse presionando el botón correspondiente del cuestionario al

finalizar la resolución.

Se recuerda que en el examen no se considera si el programa compila o no. **Las respuestas deben adecuarse de manera estricta a lo que el enunciado de cada pregunta pide.**

El enunciado para las dos primeras preguntas se encuentra en la página siguiente

Enunciado para preguntas 1 y 2

1) Una aplicación para restaurantes dispone de un conjunto de archivos para la gestión de los actividades de sus asociados, entre los que se destacan:

restaurantes.dat

Código de restaurante (entero)

Nombre (char[30])

Provincia a la que pertenece (1 a 24)

Gerente (char[30])

Categoría de restaurante (1 a 10)

bool estado;

clientes.dat

DNI (entero)

Nombre (char [30])

Medio de pago (entero entre 1 y 5)

Tipo de cliente (entero entre 1 y 10)

Edad (entero)

bool estado;

reservas.dat

Código de restaurante (entero)

DNI del cliente que reserva (entero)

Fecha de la reserva(Objeto Fecha)

Hora (entero)

bool estado;

El archivo de reservas tiene registros desde el año 2018 en adelante.

Archivos de cabecera disponibles

Resolver las preguntas considerando como si se dispusiera de un archivo de nombre **clases.h** con la definición de los siguientes clases:

```
class Fecha{  
  
    private:  
  
        int dia, mes, anio;  
  
    public:  
  
        gets() y sets()  
  
        void Cargar();  
  
        void Mostrar();  
  
};
```

```
class Restaurante{  
  
    private:  
  
        int codigoRestaurante;  
  
        char nombre[30];  
  
        int provincia;  
  
        char nombreGerente[30];  
  
        int categoriaRestaurante;  
  
        bool estado;  
  
    public:  
  
        gets() y sets()  
  
        bool leerDeDisco(int);  
  
        bool grabarEnDisco();
```

```
void Cargar();  
  
void Mostrar();  
  
};
```

```
class Cliente{  
  
    private:  
  
        int DNI;  
  
        char nombre [30];  
  
        int medioPago, tipoCliente, edad;  
  
        bool estado;  
  
    public:  
  
        gets() y sets()  
  
        bool leerDeDisco(int);  
  
        bool grabarEnDisco();  
  
        void Cargar();  
  
        void Mostrar();  
  
};
```

```
class Reserva{  
  
    private:  
  
        int codigoRestaurante;  
  
        int DNI;  
  
        Fecha fechaReserva;  
  
        int hora;  
  
        bool estado;
```

public:

gets() y sets()

bool leerDeDisco(int);

bool grabarEnDisco();

void Cargar();

void Mostrar();

};

NOTA:

bool leerDeDisco(int);/// recibe la posición del registro que se quiere leer. Devuelve true si
//puede leer y false si no puede

bool grabarEnDisco(); //agrega un registro al final del archivo. Devuelve true si pudo grabar y false
si no pudo