

## Fundamentos de Programación – Final 21/02/2017

---

**Ejercicio 1 (25 pts)** En una carrera de ciclismo por etapas, el ganador es el competidor que acumula menor cantidad de tiempo sumando sus tiempos de todas las etapas. Escriba un programa que permita ingresar la cantidad de competidores de una carrera de tres etapas y a continuación, por cada competidor, ingresar primero nombre y apellido, y luego los tres tiempos (uno por cada etapa). Cada tiempo se ingresa mediante dos enteros, que corresponden a minutos y segundos. El programa debe mostrar el nombre del ganador y su tiempo total en minutos y segundos. Advertencia: 40m 40s + 50m 50s no es 90m 90s, sino 91m 30s.

**Ejercicio 2 (30 pts)** Un struct definido como `struct Evento { string tipo; int jugador; }` representa un evento ocurrido en un partido de tenis. tipo puede ser “ace”, “error no forzado”, “tiro ganador”, “doble falta”, y “quiebre”. jugador puede ser 1 o 2. a) Escriba una función llamada `resumen_partido` que reciba un arreglo/vector de structs de tipo Evento y un número de jugador, y retorne la cantidad de eventos de cada tipo para ese jugador. b) Escriba un programa cliente que permita ingresar todos los eventos de un partido durante el mismo (sin conocer previamente la cantidad de eventos), y luego muestre en pantalla el resumen de cada jugador.

**Ejercicio 3 (25 pts)** Se pretende analizar el rendimiento académico de 200 egresados de una carrera cuyo plan contiene 35 materias. Escriba un programa que permita a) ingresar 200x35 ternas de datos compuestas por número de alumno (1 a 200), número de materia (1 a 35) y nota final (6 a 10) y muestre: b) el mejor promedio de la carrera; c) la materia más difícil (la que tenga menor promedio de nota), y d) la nota que obtuvo el alumno del mejor promedio para esa materia más difícil.

**Ejercicio 4 (20 pts)** Explique: a) ¿Qué entiende por compilación? b) En una asignación, ¿el acceso a las variables intervinientes es destructivo? c) ¿Qué significa que una estructura de datos sea homogénea? Mencione un ejemplo de una estructura que lo sea, y otro de una que no. d) ¿Qué significa que una función sea recursiva? ¿Tendría sentido plantear una función recursiva sin utilizar estructuras de control condicionales?

**Ejercicio Libres (20 pts)** Escriba una función que reciba un arreglo `v` de enteros ordenados de mayor a menor, y un entero adicional `x`. La función debe insertar el elemento `x` en el arreglo `v`, solamente si el elemento no estaba en el mismo (si ya estaba no debe hacer nada), y en la posición adecuada para que el arreglo se mantenga ordenado de mayor a menor luego de la inserción. La función debe retornar un `bool` indicando si realizó o no la inserción.

**Ejercicio 1 (25pts)** Escriba un programa en C++ que simule un sistema de control de acceso por contraseña de la siguiente manera: a) En una primer etapa debe permitir cargar los datos de los usuarios válidos (pares de datos: nombre y contraseña). No se conoce la cantidad de usuarios válidos y los datos terminan con el usuario "ZZZ". En esta carga inicial, debe verificar mediante un función validar\_contraseña que la contraseña no sea "1234", "asdf", "pass" ni la cadena vacía; mostrando un mensaje de error y solicitando el reingreso de la misma en estos casos. (nota: proponga usted el prototipo de la función e impleméntela). b) Luego, simular el verdadero intento de acceso al sistema. El programa debe mostrar el mensaje "Por favor, identifíquese:" y solicitar nombre y contraseña. Si se ingresa un par nombre+contraseña existente (que coincida con los cargados en el apartado a) se debe mostrar el mensaje "Bienvenido al sistema sr X" (reemplazando X por el nombre del usuario); sino, luego de 5 intentos fallidos debe mostrar el mensaje "Fuera bicho!".

**Ejercicio 2 (25pts)** La Fórmula 1 está buscando mejorar la seguridad agregando nuevos elementos para proteger las cabezas de los pilotos. Se probaron tres sistemas: el "shield" (una especie de "parabrisas" blindado), el "halo" (un armazón de carbono) y el "ghost" (sin piloto en el auto, manejado con control remoto). Ciertos equipos han probado algunos de estos elementos y han calificado los resultados en una escala de 1 a 5 estrellas. Se requiere un programa C++ para analizar estos datos y ayudar a decidir cuál es la mejor opción. a) Se ingresan ternas de datos compuestas por un código de equipo (de 1 a 10), un código de sistema (1-shield, 2-halo, 3-ghost), y la calificación (1 a 5 estrellas). Los datos finalizan con código de equipo 0. Organice estos datos en una matriz de 10x3 (10 equipos, 3 sistemas), donde cada posición debe guardar la cantidad de estrellas asignadas por ese equipo a ese sistema, o -1 si ese equipo no probó ese sistema. b) muestre solo los equipos que calificaron los tres sistemas, con sus respectivas calificaciones. c) Informe el nombre del sistema que obtuvo un mayor promedio de estrellas (el promedio se calcula considerando todos los equipos que probaron el sistema, sin considerar los -1). Defina y utilice para ello una función calcular\_promedio que calcule el promedio de un sistema.

**Ejercicio 3 (30pts)** a) Defina un struct Licencia para representar una licencia de un software. El struct debe tener campos para guardar el nombre del software (ejemplo: "Zinjal"), el nombre de licencia (ejemplo "GPL") y el costo de la misma en dólares (ej: 0). b) Implemente una función calcular\_costos que reciba dos vectores, uno con una lista de licencias, y otro de igual tamaño con una lista de enteros, que indican las cantidades de cada una de las licencias que necesita adquirir cierta institución. La función debe calcular y retornar el costo total de la compra de todas esas licencias. c) Implemente un programa cliente que permita cargar los datos de N programas (por cada uno, su licencia y la cantidad a comprar) y calcule y muestre el costo total.

**Ejercicio 4 (20pts)** a) ¿Qué condiciones debe cumplir una función para que sea recursiva? Proponga un ejemplo (implementación completa). b) ¿Qué significa pasaje de parámetros por valor y por referencia? ¿Cuándo utilizaría cada uno? c) ¿Qué entiende por sobrecarga de funciones? ¿cómo se determina cuál corresponde para una determinada llamada en el programa cliente? d) ¿Qué significa que una estructura de datos sea homogénea? Mencione una que lo sea y una que no lo sea. e) ¿Cómo se define en C++ el ámbito de validez de una variable?

**Libres 15pts)** Escriba una función inserta\_col que reciba una matriz A de NxM doubles, un vector V de M doubles, y un entero i entre 0 y M-1. La función debe modificar la matriz A, insertando una nueva columna i-ésima colocando en ella los valores del vector V.

## Fundamentos de Programación – Final 19/12/2017

---

**Ejercicio 1 (25 pts)** Un triatleta, preparándose para una competencia importante, ha entrenado durante varias semanas en el mismo circuito donde se realizará dicha competencia y registrado en cada entrenamiento sus tiempos para cada uno de los tres tramos (natación, ciclismo, pedestrismo). a) Escriba un programa que permita ingresar N grupos de 7 datos de datos conformados por la fecha (como un solo número aaaammdd) y los tres tiempos (cada tiempo será ingresado mediante dos números: minutos y segundos). El programa luego deberá informar: b) por cada disciplina, el mejor (menor) tiempo logrado y la fecha del mismo; c) el promedio de los tiempos totales (sumando las tres disciplinas); y d) los dos mejores tiempos totales y la fecha en que se realizaron.

**Ejercicio 2 (30 pts)** a) Defina un struct Ejercicio para representar un ejercicio de un examen. El struct debe almacenar un nombre de ejercicio (una palabra que lo identifica), el texto del enunciado, el puntaje que otorga y el nivel de dificultad (valor real, de 1 a 5). b) Escriba una función, que dado un arreglo de Ejercicios y un nombre, busque y retorne la posición del ejercicio con ese nombre en el arreglo (nota: los nombres son únicos, no habrá dos ejercicios con el mismo nombre). Si no hay ninguno con ese nombre, debe retornar -1. c) Escriba un programa cliente que permita cargar un vector de N ejercicios con todos los ejercicios que alguna vez se tomaron en la materia (para esta primera carga, N es dato), y luego ingresar nombres de ejercicios para confeccionar un nuevo examen combinando ejercicios del vector (esta segunda entrada finaliza cuando el nombre se deja en blanco). Por cada nombre ingresado, debe utilizar la función del apartado b para verificar que existe y obtener sus datos. Cuando la carga finalice deberá mostrar la dificultad promedio del examen generado (promediando las dificultades de los ejercicios seleccionados), y avisar con un mensaje alusivo si los puntajes seleccionados no suman 100.

**Ejercicio 3 (25 pts)** Escriba una función que reciba una matriz de flotantes, y dos números de filas (f1 y f2), y reemplace todas las filas desde f1 hasta f2 (incluyendo ambas) por una única fila conformada en cada columna por el promedio de los datos eliminados de esa columna.

**Ejercicio 4 (20 pts)** Explique: a) ¿Qué significa que una función sea “recursiva”? ¿Puede existir y tener sentido una función recursiva que contenga solamente instrucciones secuenciales? (en caso afirmativo, ejemplifique) b) ¿Cuál es la diferencia entre pasaje por valor y pasaje por referencia? ¿Cuándo utilizaría uno u otro? c) Explique qué significa sobrecarga de funciones y cómo detecta el compilador cuál función debe ejecutar ante una llamada en el caso de sobrecarga. d) ¿Qué significa “compilar” un programa? ¿Podría ejecutar un programa sin antes compilarlo?

**Libres (20 pts)** Escriba una función que reciba un arreglo v de enteros ordenados de mayor a menor, y un entero adicional x. La función debe insertar el elemento x en el arreglo v, solamente si el elemento no estaba en el mismo (si ya estaba no debe hacer nada), y en la posición adecuada para que el arreglo se mantenga ordenado de mayor a menor luego de la inserción. La función debe retornar un bool indicando si realizó o no la inserción, y un entero indicando la posición en la que se insertó/encontró el valor.

## Fundamentos de Programación 2017 – 2do Parcial - TEMA A

---

**Ejercicio 1 (25 pts).** Se ingresan 200 valores numéricos provenientes de un sensor de presión. Debido a fallas de comunicación, se ingresaron algunos valores negativos. Realice un programa que reemplace todos los valores negativos por el promedio entre el máximo y mínimo valor lista (tenga en cuenta de no considerar los valores negativos en la obtención del menor). Muestre el arreglo modificado y informe cuántos reemplazos hizo.

**Ejercicio 3 (20 pts).** Se ingresan como datos N valores numéricos ordenados en forma descendente (el valor de N se lee por teclado). Realice un algoritmo que lea un nuevo valor y lo inserte en la lista de tal manera que el orden se mantenga. Luego muestre la nueva lista resultante.

**Ejercicio 2 (35 pts).** Una empresa que comercializa jugos en caja, tiene 15 vendedores quienes que levantan pedidos de clientes durante el mes de junio (30 días). Por cada pedido, se ingresan sin orden alguno las ternas correspondientes al código del vendedor (1 a 15), el día (1...30) y la cantidad de cajas de jugo encargadas, pudiendo haber más de un pedido diario del mismo vendedor. Los datos finalizan con el código de vendedor igual a 0. Luego, se desea obtener e informar:

- a) ¿Cuál es el código de vendedor que más pedidos hizo para el día 10?
- b) ¿Cuál es la cantidad vendida por día considerando todos los vendedores?
- c) ¿Cuántas cajas vendió en total el vendedor 10?

**Ejercicio 4 (20 pts).** Explique/justifique: a) ¿Es posible guardar en un arreglo datos de tipos distintos? Justifique su respuesta. b) ¿Es indistinto utilizar cualquiera de las estructuras iterativas: “Mientras” y “Repetir Hasta Que”? Explique. c) ¿Qué acciones/instrucciones en pseudocódigo permiten inicializar/definir variables? d) ¿Mencione cuáles son las etapas de resolución de un problema?

## Fundamentos de Programación 2017 – 2do Parcial - TEMA B

---

**Ejercicio 1 (25 pts).** En una carrera de 200 metros participaron N atletas, donde N es un valor que se lee al comienzo (se sabe que hay menos de 50 atletas). Por cada atleta se ingresa el nombre y tiempo registrado (en seg). a) Muestre los nombres de los 2 ganadores y sus tiempos. b) Determine cuántos atletas estuvieron a 5 segundos o menos de obtener el segundo lugar.

**Ejercicio 2 (35 pts).** La ciudad asigna un monto fijo de \$100.000 por cada mes, a cada una de 10 escuelas rurales durante todo el 2017 para gastos. Mensualmente cada escuela realiza compras que se irán acumulando. Escriba un programa que organice una tabla de 10x12 con los totales de montos que cada escuela ha tenido en cada mes. Los datos que se ingresan por cada compra son: código de de la escuela (entre 1 y 10), día, mes, año y el monto, pudiendo haber más de una compra mensual para la misma escuela. Los datos finalizan con el código de escuela igual a 0. Luego, se desea obtener e informar: a) Al finalizar el año, ¿cuánto dinero le sobró a cada escuela? b) ¿Cuántos escuelas gastaron todo el dinero en el mes de febrero? c) ¿Cuál fue la escuela que menos gastó en octubre?

**Ejercicio 3 (20 pts).** Se ingresan como datos 100 nombres de personas que están ordenados alfabéticamente. Realice un algoritmo que lea un nuevo nombre y lo inserte en la lista de tal manera que el orden se mantenga. Luego muestre la nueva lista resultante.

**Ejercicio 4 (20 pts)** Explique/justifique: a) ¿Cuáles son las diferencias de las estructuras iterativas Mientras y Repetir? b) Al ejecutarse la acción de asignación, ¿el acceso a las variables intervinientes es destructivo? (¿se pierden sus valores?) c) Mencione las etapas de la resolución de problemas. d) ¿Qué es un arreglo y cuáles son sus características?

## Fundamentos de Programación 2017 – Recuperatorio 2do Parcial

---

**Ej. 1 (25pts)** En un programa C++ se ha definido el tipo: `struct datetime {int hh, mm, ss};` donde 3 enteros representan una hora determinada (hora, minutos y segundos). Escriba el código de una función C++ llamada `dif_time(...)` de tipo `datetime`, la cual debe recibir como parámetros 2 variables de tipo `datetime`. La función debe calcular la diferencia de tiempo entre el 1er parámetro y el 2do y retornarla también en horas, minutos y segundos mediante `datetime`. Escriba un programa cliente de la función `dif_time()` que ingrese 2 horarios y muestre la diferencia de tiempo. Sugerencia: en la función, convierta ambos horarios a solo segundos, calcule la diferencia en segundos, y convierta luego el resultado a horas, minutos y segundos para retornarlo.

**Ej. 2 (35pts)** Escriba un programa C++ que lea inicialmente por teclado los presupuestos anuales asignados a 10 facultades de una universidad al inicio del año 2016 (10 valores). Luego debe leer ternas correspondientes a cada gasto efectuado por la facultad en el 2016: código de facultad, mes, monto. El programa debe organizar una tabla de 10 filas (facultades) por 12 columnas (meses) donde se indiquen en cada celda el total gastado en ese mes. En un mismo mes, una misma facultad puede realizar varios gastos. Determinar: a) el saldo final anual (lo que quedó sin gastar) de cada facultad; b) Cuántas facultades finalizaron el año con saldo negativo (déficit); c) La facultad y el monto correspondiente al mayor gasto mensual de toda la tabla. Completar los ítems a, b y c usando funciones.

**Ej. 3 (20pts)** a) Proponga 2 ejemplos de funciones recursivas. b) Proponga solo el prototipo de la función `min_max(..)`, que reciba un arreglo de enteros como parámetros y devuelva el mayor y el menor del arreglo sin emplear el operador `&`.

**Ej. 4 (20pts)** a) (2 pts) Señale las diferencias entre programa fuente y código ejecutable. b) (3 pts) Explique qué significa sobrecarga de funciones y cómo detecta el compilador cuál función debe ejecutar ante una llamada en el caso de sobrecarga. c) (3 pts) ¿Siempre debe usar el operador `&` para pasaje por referencia en funciones? Justifique su respuesta. d) (3 pts) Mencione al menos 3 diferencias entre arreglos y structs e) (3 pts) ¿Siempre es posible reemplazar una estructura `while` por otra de tipo `for`? Explique por qué es posible o dé un contraejemplo. f) (3 pts) ¿Cuáles son las condiciones para que una función sea recursiva? g) (3 pts) Mencione al menos 3 ventajas de utilizar funciones.

## Fundamentos de Programación 2016 - 2do Parcial - TEMA A - 21/06/2016

---

**Ejercicio 1 [25 pts]** Escriba un programa que permita leer una serie de valores numéricos enteros y los almacene en un arreglo. Se desconoce de antemano de cuántos valores se trata, pero se sabe que la lectura de datos termina cuando se ingresa el valor -1. Escriba una función que cuente la cantidad de valores que se encuentran por encima del promedio e inserte dicha cantidad al principio del arreglo. Utilizar la función desde el programa principal y mostrar el arreglo antes y después de la inserción.

**Ejercicio 2 [25 pts]** Escriba un programa que lea los datos de 30 alumnos que asistieron a una evaluación de programación, por cada uno de ellos se ingresa: cod. alumno, dni, nombre y nota. Los datos llegan sin orden alguno. Una vez terminada la lectura, el usuario deberá ingresar el código correspondiente uno de los alumnos ingresados y el programa deberá mostrar todos sus datos. Utilizar al menos 2 funciones.

**Ejercicio 3 [30 pts]** Se desea escribir un programa que permita gestionar puntos de venta para pasajes de colectivos durante el año 2016. Existen 10 puntos de venta y se desea conocer la cantidad de pasajes vendidos mensualmente. El programa debe recibir como entrada ternas desordenadas de datos: cód de punto de venta (1-10), fecha (con formato aaaammdd) y cantidad de pasajes vendidos. Se desea conocer: a) cuántos pasajes vendió el punto de venta 5 durante todo el año, b) cuántos pasajes se vendieron, teniendo en cuenta todos los puntos de venta, en un determinado mes ingresado por el usuario, c) la cantidad puntos de venta no vendieron pasajes durante el mes de octubre. Resolver al menos 1 de los 3 apartados empleando una función.

**Ejercicio 4 [20 pts]** Responda: a) Mencione al menos 3 reglas para nombrar variables en C++. b) ¿Es posible almacenar valores enteros en una variable de tipo float? ¿Y viceversa? Explique. c) ¿Qué diferencias encuentra entre un struct y un array? ¿Cuándo conviene emplear cada uno? d) Explique las diferencias entre pasaje de parámetros por valor y por referencia. e) ¿Cuáles son las condiciones para que una función sea recursiva?

**Ejercicio 1 [25 pts]** Escriba un programa que permita ingresar al usuario 30 valores numéricos que deberán almacenarse en un arreglo. Escriba una función que encuentre y devuelva los 2 valores mayores del arreglo y utilice dicha función desde el programa principal.

**Ejercicio 2 [30 pts]** Se desea escribir un programa que permita gestionar una red de verdulerías. Existen 10 puntos de verdulerías diferentes y cada una comercializa 30 productos distintos y se necesita conocer la cantidad de productos vendidos por cada verdulería. El programa debería leer inicialmente el código de cada producto y su descripción (por ejemplo: 0, zanahorias. Luego se deberían leer ternas ordenadas con los siguientes datos: cod. producto, cod. verdulería, cantidad de unidades vendidas. Éstos datos deben disponerse en una matriz de 10 filas por 30 columnas. Luego: a) Agregar una fila adicional con los totales vendidos de cada producto teniendo en cuenta todas las verdulerías. b) Mostrar la descripción del producto que registró más ventas en la en la verdulería 5. c) La cantidad de productos con código 3 que se vendieron en la verdulería con código 5. **Importante: al menos el apartado a) ó el apartado b) (queda a elección del alumno) deben resolverse utilizando una función.**

**Ejercicio 3 [25 pts]** Escriba un programa que lea los datos de 20 atletas que asistieron a una maratón, por cada uno de ellos se ingresa: cod. participante, dni, nombre y tiempo que tardó en cruzar la meta. Los datos llegan sin orden alguno. Una vez terminada la lectura, el usuario deberá ingresar el dni correspondiente uno de los participantes ingresados y el programa deberá mostrar todos sus datos. Utilizar al menos 2 funciones.

**Ejercicio 4 [20 pts]** Responda: a) En un lenguaje de programación, ¿qué se entiende por expresión? b) ¿Qué diferencia hay entre definir e inicializar una variable? c) Explique qué significan parámetros formales y parámetros actuales. d) ¿Es posible asignar el resultado de una expresión a una variable de tipo diferente al tipo del resultado de la expresión? Explique. e) Considere la siguiente expresión: cout<<(x=23), ¿tiene sentido? Explique



**Ej. 1 (20pts)** En un programa C++ se ha definido el tipo: `struct datatime {int hh, mm, ss};` donde 3 enteros representan una hora determinada (hora, minutos y segundos). Escriba el código de una función C++ llamada `diff_time(...)` de tipo `datatime`, la cual debe recibir como parámetros 2 variables de tipo `datatime`. La función debe calcular la diferencia de tiempo entre el 1er parámetro y el 2do.

**Ej. 2 (35pts)** En una localidad se realizan elecciones municipales. Se presentan 7 partidos políticos y existen 18 barrios en el municipio. Escriba un programa en C++ que lea inicialmente los nombres de cada candidato y el código de partido (1..7) al que pertenece. Luego debe leer los datos que llegan de cada una de las 145 mesas electorales. Se ingresan 9 datos por mesa: cod. barrio, cant. votos blanco, cant. votos partido 1, cant. votos partido 2, ..., cant. votos partido 7. a) Escriba una función que permita obtener el código del partido ganador y el % de votos obtenido sobre el total. b) Informe el nombre del próximo intendente. c) Muestre el % de votos obtenido por cada candidato. d) Escriba una función que devuelva el cod. del barrio que obtuvo mayor cantidad de votos en blanco.

**Ej. 3 (25pts)** En una prueba de maratón compiten 35 atletas. Organice un struct con los datos de cada atleta: apellido, nombres, número asignado, h1,m1,s1; donde número asignado es un entero entre 1 y 35; y h1,m1,s1 son 3 enteros para el tiempo empleado (hora, minutos, segundos) en completar la prueba. Si tiempo empleado es 0,0,0 significa que el atleta abandonó la prueba. Escriba un programa C++ que ingrese -una vez terminada la competencia- los datos, sin orden, de cada uno de los 35 atletas. a) Escribir una función que determine el atleta ganador y mostrar sus datos desde el programa cliente. b) Luego, mostrar los apellidos y nombres de todos los atletas que terminaron la competencia (sin contar los que abandonaron).

**Ej. 4 (20pts)** a) (2 pts) Señale las diferencias entre programa fuente y código ejecutable. b) (3 pts) Explique qué significa sobrecarga de funciones y cómo detecta el compilador cuál función debe ejecutar ante una llamada en el caso de sobrecarga. c) (3 pts) ¿Siempre debe usar el operador `&` para pasaje por referencia en funciones? Justifique su respuesta. d) (3 pts) Mencione al menos 3 diferencias entre arreglos y structs e) (3 pts) ¿Siempre es posible reemplazar una estructura `while` por otra de tipo `for`? Explique por qué es posible o dé un contraejemplo. f) (3 pts) ¿Cuáles son las condiciones para que una función sea recursiva? g) (3 pts) Mencione al menos 3 ventajas de utilizar funciones.

**Ej 1 (20pts)** Considere el struct fecha{unsigned int anio, mes, dia;} y escriba el código de una función C++ llamada calcular\_edad(...) que devuelva un entero con la edad de una persona en años y debe recibir 2 parámetros de tipo fecha: el 1ro con la fecha de nacimiento de una persona y el 2do parámetro con la fecha actual. La edad debe corresponder a años cumplidos.

**Ej. 2 (25pts)** Defina en un programa C++ el struct persona {unsigned int dni, anio\_n, mes\_n, dia\_n, edad; string apellido, nombres;} donde anio\_n, mes\_n y dia\_n representan el anio, mes y día de nacimiento de la persona. Luego ingrese los datos de n personas. a) Muestre Apellido, Nombres, Edad, DNI de todas las personas cuya edad supere el promedio de edades de todos los datos ingresados. b) Escriba una función que reciba un apellido y elimine a la primer persona con dicho apellido. Luego muestre, desde el programa cliente, los datos antes y después de la eliminación.

**Ej. 3 (35pts)** Se ingresan como datos en un programa C++ las producciones anuales de soja, maíz y trigo (en miles de Tn) de 8 provincias en el decenio 2005-2014. Para ello se leen 240 gupos de datos: CodProv, Anio, Cant y TipoCultivo; donde CodProv es un entero entre 1 y 8 correspondiente a la provincia, Anio un entero entre 2005 y 2014 y Cant la cantidad de Tn producidas y TipoCultivo es un string ("soja", "trigo" o "maíz"). a) Determine cuáles provincias tuvieron una producción menor de trigo en 2014 respecto de 2005. b) Escriba una función que reciba un entero correspondiente a un año y devuelva cuál fue provincia produjo más soja en dicho año. Utilice dicha función para obtener la provincia que produjo mayor cantidad de soja en el 2008. c) Informe el total de maíz producidos por las 8 provincias en 2013.

**Ej. 4. (20pts)** a) (3 pts) Indique 2 formas de obtener varios resultados en una llamada a función. b) (2 pts) Es posible utilizar constantes como parámetros actuales (de llamada) a una función. c) (3 pts) ¿Que entiende por ámbito de validez de variables en C++? d) (3 pts) Explique para qué sirve la sentencia break y por qué su uso no cumple con las reglas de la programación estructurada. e) (3 pts) ¿Cuáles son las condiciones para que una función sea recursiva? f) (3 pts) Mencione al menos 3 diferencias entre arreglos y structs. g) (3 pts) ¿Es posible que desde una función se devuelva un arreglo con la sentencia return? ¿Y un struct? Explique qué ocurre en cada caso.

## **Fundamentos de Programación - Exámen Final 05/07/16**

**Ejercicio 1 [15 pts]** Escriba una función que reciba una matriz de elementos de tipo float de 20 filas y 14 columnas, y genere un vector con los mayores elementos de cada fila.

**Ejercicio 2 [35 pts]** Un negocio necesita un programa para automatizar la gestión del stock, que permita primero cargar el stock inicial de productos, luego ingresar la ventas realizadas durante un período, y finalmente muestre aquellos productos para los cuales el stock ha quedado debajo de cierto límite luego de dichas ventas. Escriba un programa C++ que permita:

- a) Ingresar primero ternas de datos formadas por código, descripción, y cantidad. Cada terna indica el stock inicial de un cierto producto que se identifica por su código. Se sabe que el negocio vende 200 productos diferentes (los códigos van de 1 a 200).
- b) Luego, el programa debe permitir ingresar las ventas: se leen pares formados por código y cantidad de la venta de cada producto. Estos pares finalizan cuando llega el código de producto 0.
- c) Finalmente, el programa debe permitir ingresar cantidades enteras X, y por cada cantidad X ingresada mostrar la lista de productos que ha quedado con menos de X unidades en stock luego de las ventas. Cuando X sea 0, el programa deberá finalizar.

**Ejercicio 3 [30 pts]** Se desea seleccionar dos alumnos de una lista de 30 candidatos para una pasantía. Escriba un programa que permita ingresar los datos de los 30 alumnos, los guarde en un vector de structs, y obtenga mediante una función los apellidos y nombres de los dos mejores. Por cada alumno se ingresa apellido y nombre, promedio, cantidad de materias aprobadas, y cantidad de aplazos. Defina un struct para agrupar estos datos. Para esta pasantía se considera a un alumno mejor que otro si tiene mayor cantidad de materias aprobadas. En caso de empate, se considera mejor a quien tenga menor cantidad de aplazos. Si nuevamente hay empate (igual cantidad de materias aprobadas y de aplazos), a quien tenga mejor promedio.

**Ejercicio 4 [20 pts]** Explique: a) En una operación de asignación, el acceso a las variables ¿es destructivo? b) ¿Qué significa “sobrecarga de funciones”? c) ¿Qué entiende por “scope” o “ámbito de validez” de una variable? ¿Cómo se determina en C++? d) Defina “intérprete” y “compilador” señalando ventajas y desventajas de cada uno.

**Ejercicio Libre [20 pts]** Escriba una función “buscar” recursiva para buscar un elemento en un vector desde una posición que se recibe como argumento. Por ejemplo, si v es un vector de *strings* y se invoca como *buscar(v,5,”hola”)*, la función debe buscar un elemento del vector igual a “hola” desde la posición 5 hasta el final del mismo. La función debe retornar la posición en donde se encuentra el elemento, o -1 en caso de que el elemento no se encuentre.

## UNL-FICH - Fundamentos de Programación - Examen Final - 21/02/17

**Ejercicio 1 (25 pts)** En una carrera de ciclismo por etapas, el ganador es el competidor que acumula menor cantidad de tiempo sumando sus tiempos de todas las etapas. Escriba un programa que permita ingresar la cantidad de competidores de una carrera de tres etapas y a continuación, por cada competidor, ingresar primero nombre y apellido, y luego los tres tiempos (uno por cada etapa). Cada tiempo se ingresa mediante dos enteros, que corresponden a minutos y segundos. El programa debe mostrar el nombre del ganador y su tiempo total en minutos y segundos. Advertencia: 40m 40s + 50m 50s no es 90m 90s, sino 91m 30s.

**Ejercicio 2 (30 pts)** Un struct definido como *struct Evento { string tipo; int jugador; }* representa un evento ocurrido en un partido de tenis. *tipo* puede ser “ace”, “error no forzado”, “tiro ganador”, “doble falta”, y “quiebre”. *jugador* puede ser 1 o 2. a) Escriba una función llamada *resumen\_partido* que reciba un arreglo/vector de structs de tipo *Evento* y un número de jugador, y retorne la cantidad de eventos de cada tipo para ese jugador. b) Escriba un programa cliente que permita ingresar todos los eventos de un partido durante el mismo (sin conocer previamente la cantidad de eventos), y luego muestre en pantalla el resumen de cada jugador.

**Ejercicio 3 (25 pts)** Se pretende analizar el rendimiento académico de 200 egresados de una carrera cuyo plan contiene 35 materias. Escriba un programa que permita a) ingresar 200x35 ternas de datos compuestas por número de alumno (1 a 200), número de materia (1 a 35) y nota final (6 a 10) y muestre: b) el mejor promedio de la carrera; c) la materia más difícil (la que tenga menor promedio de nota), y d) la nota que obtuvo el alumno del mejor promedio para esa materia más difícil.

**Ejercicio 4 (20 pts)** Explique: a) ¿Qué entiende por compilación? b) En una asignación, ¿el acceso a las variables intervinientes es destructivo? c) ¿Qué significa que una estructura de datos sea homogénea? Mencione un ejemplo de una estructura que lo sea, y otro de una que no. d) ¿Qué significa que una función sea recursiva? ¿Tendría sentido plantear una función recursiva sin utilizar estructuras de control condicionales?

**Ejercicio Libres (20 pts)** Escriba una función que reciba un arreglo *v* de enteros ordenados de mayor a menor, y un entero adicional *x*. La función debe insertar el elemento *x* en el arreglo *v*, solamente si el elemento no estaba en el mismo (si ya estaba no debe hacer nada), y en la posición adecuada para que el arreglo se mantenga ordenado de mayor a menor luego de la inserción. La función debe retornar un *bool* indicando si realizó o no la inserción.

## **FICH-UNL. Fundamentos de Programación. 23-06-2015. Recuperatorio Parcial 2.**

**Ej. 1. (35 pts).** Una empresa constructora lleva un registro del trabajo de sus 50 empleados. De cada empleado almacena nro. de empleado (1..50), nombre y apellido, CUIT y la fecha de nacimiento (AAAAMMDD). De los trabajos realizados durante 12 meses se desea llevar un registro del salario mensual que percibió cada empleado. Para esto primeramente se realiza la lectura de datos de los 50 empleados en forma desordenada. Posteriormente, para la carga de los salarios, se ingresa el número de empleado, el mes y el salario que percibió. Se le solicita que realice la lectura e implemente lo siguiente: **a).** Una función *mejor\_pago()*, que retorne el nro. de empleado que recibió el mayor salario para un determinado mes que es leído por teclado y luego, lo muestre en pantalla desde el programa principal. **b).** Una función que reciba como argumento la fecha actual (AAAAMMDD) y devuelva al programa principal la cantidad de empleados mayores y menores a 50 años. **c).** Una función que retorne el mes donde se pagó menos dinero en salarios.

**Ej. 2. (25 pts):** El juego del galeón consiste en realizar tiros intentando acertar en una celda donde se encuentre un barco. El campo de juego es una matriz de 10x10, cuyos valores son 0 para el agua y 1 para los barcos, teniendo en cuenta que un barco corresponde a una única celda. Un barco es hundido al reemplazar el valor 1 de una celda por 0. **a)** Usando una función genere N barcos en posiciones aleatorias, teniendo en cuenta de no sobrescribir un barco ya existente. La función también recibe como argumento la cantidad de barcos a generar. **b)** Implemente una función que retorne la cantidad de barcos que quedan sin hundir. **c)** Simule tantos disparos como sea necesario hasta que todos los barcos sean hundidos. **d)** Muestre la cantidad de disparos fallidos y el porcentaje de aciertos (aciertos/disparos).

**Ej. 3. (20 pts).** Se lee en un arreglo lineal de 10 elementos nombres de personas ya ordenados alfabéticamente, realice la lectura y programe lo siguiente: **a).** Permita ingresar un nombre a ser eliminado, para esto debe desplazar los nombres pisando el nombre a eliminar. **b).** Inserte un nuevo nombre manteniendo el orden alfabético del arreglo.

**Ej. 4 (20 pts).** **a)** Señale las diferencias entre programa fuente y código ejecutable. **b).** Explique qué significa que una función sea recursiva y qué condiciones son necesarias para su uso. **c).** Explique en qué consiste el pasaje de parámetros por referencia. **d).** Mencione características y limitaciones de arreglos estáticos. **e).** ¿Es correcta la sobrecarga del prototipo de la función calcula? Justifique.

```
float calcula (int a, int b);
```

```
int calcula(int a, int b);
```

## Fundamentos de programación - Final 07/07/2015

**Ejercicio 1** (25 pts) a) Declare un struct *DatosMatriz* que contenga los siguientes campos: un flotante *promedio*, y dos enteros *cantidad\_menor* y *cantidad\_mayorIgual*. b) Escriba una función *AnalizarMatriz* que reciba una matriz de Nx12 valores flotantes y retorne un struct del tipo *DatosMatriz*. En el struct debe guardar el promedio de todos los valores de la matriz, y las cantidades de elementos de la misma mayores o iguales y menores a dicho promedio. c) Escriba un programa cliente que genere una matriz de 5x12 flotantes con valores aleatorios entre 0 y 1, y muestre los resultados de invocar a la función con dicha matriz. *Ayuda: para generar un flotante aleatorio entre 0 y 1 puede usar la expresión “(rand()%1001)/1000.f”.*

**Ejercicio 2** (30 pts) En una carrera de motos se registraron los tiempos de cada una de las 27 vueltas de cada uno de los 18 competidores, y ahora se desean analizar estos datos. Escriba un programa para: a) Permitir *primero* la carga de los nombres de cada uno de los competidores (en orden), y *luego*, (también en orden) los tiempos de cada competidor por cada vuelta (en segundos). b) Encontrar la mejor vuelta de toda la carrera e informar su tiempo, qué número de vuelta fue, y el nombre del competidor que la logró (nota: la mejor es la de *menor* tiempo). c) Informar **el tiempo de vuelta promedio** considerando los 18x27 tiempos (el promedio de todos los tiempos). *Debe utilizar funciones para los incisos b y c.*

**Ejercicio 3** (25 pts) Se desea determinar cuáles fueron los dos delanteros más efectivos de la copa América. Escriba un programa para resolver este problema de la siguiente forma: a) El programa debe permitir ingresar por cada delantero seis datos: apellido nombres, cantidad de minutos jugados (MJ), goles convertidos (GC), tiros al arco (TA), tiros desviados (TD) y pases de gol realizados (PG). La carga de datos finaliza cuando el apellido se deja en blanco (cuando se presiona enter sin ingresar nada). b) El programa debe calcular, para cada delantero, un coeficiente de **efectividad** que se obtiene como  $(2*GC+PG)/MJ$ , y otro para la mala puntería denominado **badaiming** que se obtiene como la división entre los tiros desviados (TD) y el total de tiros efectuados al arco (TA). c) Finalmente, debe informar los apellidos de los dos delanteros con mayores valores de **efectividad**. Si hay dos delanteros con la misma **efectividad**, se considera mejor al que tenga un menor índice de **badaiming**.

**Ejercicio 4** (20 pts) **Explique** (“si” y “no” no son respuesta válidas si no incluyen una breve justificación) a) ¿Qué significa que una función sea “recursiva”? ¿Puede existir y tener sentido una función recursiva que contenga solamente instrucciones secuenciales? (en caso afirmativo, ejemplifique) b) ¿Cuál es la principal diferencia entre las estructuras repetitivas Mientras-Hacer (while{...}) y Repetir-Hasta (do{...}while)? ¿Siempre se puede reemplazar una por la otra? c) ¿Qué significa “compilar” un programa? ¿Podría ejecutar un programa sin antes compilarlo? d) Defina los conceptos: identificador, variable, constante y expresión. e) ¿La asignación a una variable es una operación destructiva, o se conserva el valor de la misma?