

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

FUNDAMENTOS DE PROGRAMACIÓN

TERCERA EVALUACIÓN – II TÉRMINO 2012-2013

Nombre: _____ Matrícula: _____ Paralelo: _____

TEMA 1 (25 pts)

En cierta institución bancaria una inversión consta de: un Capital m por el cual se recibe un x interés anual. El problema consiste en determinar el capital que se tendrá al cabo de n años.

Ejemplo:

Supóngase que $m = \$5000.00$ y $x = 10\%$, entonces:

```
C0 = 5000 // Capital Inicial
C1 = C0 + C0 * 10% // Capital al finalizar el primer año de inversión
C2 = C1 + C1 * 10% // Capital al finalizar el segundo año de inversión
C3 = C2 + C2 * 10% // Capital al finalizar el tercer año de inversión
...
Cn = Cn-1 + Cn-1 * 10% // Capital al finalizar el ultimo año de inversión
```

Implemente una **función recursiva** en C, para calcular el capital final a recibir para cualquier: capital inicial, interés y número de años.

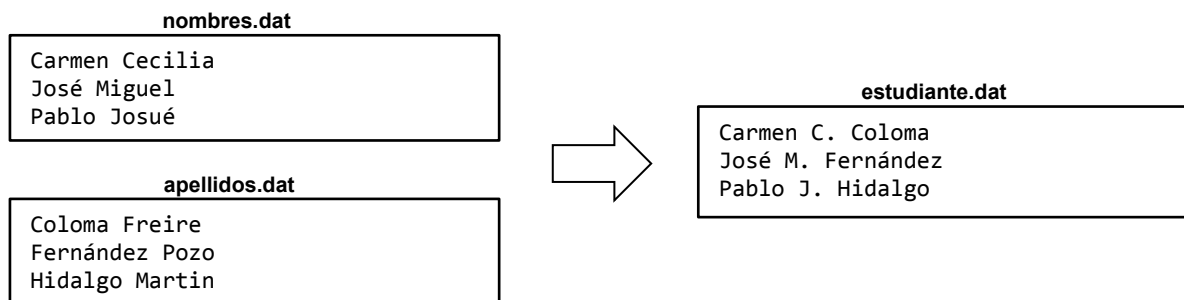
TEMA 2 (30 pts)

El cierto sistema informático el registro de los estudiantes se maneja de forma tal que los nombres y apellidos están en archivos separados, se ha solicitado que estos estén juntos en un tercer archivo pero de *forma abreviada*, es decir, que solo se registre el primer nombre seguido de la inicial del segundo nombre y el apellido paterno.

Implemente un procedimiento que realice lo solicitado, respetando el siguiente prototipo:

```
void merge(FILE *pNom, FILE *pApe, FILE *pEst);
```

Los archivos donde se almacenan los nombres y apellidos tienen el siguiente formato:



TEMA 3 (45 pts)

Las pilas son estructuras tipo LIFO (last in – first out) con una amplia aplicación en la programación. Poseen funciones para agregar (push) y remover (pop) elementos, así como para consultar el estado de la pila si esta vacía o llena. Las pilas pueden implementarse usando arreglos para lo cual se tiene que usar una variable tipo índice que muestre donde esta el último elemento que ingreso a la pila, ese índice es conocido como el tope de la pila (peak). Además el arreglo podrá almacenar un número máximo de elementos (MAX). Para esta implementación peak deberá ser declarada de forma global.

A usted se le ha solicitado implementar:

- (5 pts) El procedimiento `void init(int stack[])` que inicializa una pila para ser usada más adelante, la inicialización consiste en llenar el arreglo `stack` con valores de 0 y establecer el valor de la variable global `peak` en `-1`, para indicar que no hay ningún numero almacenado en la pila, es decir, esta vacía.

- b) (5 ptos) La función `int isEmpty(int stack[])`, permite saber si la pila esta vacía y retornará **1** si el valor la variable global **peak** es negativo lo cual significa que no hay ningún numero almacenado y si **peak** no es negativo retornará **0** indicando que no esta vacía.
- c) (5 ptos) La función `int isFull(int stack[])`, permite saber si la pila esta llena y retornará **1** si el valor de la variable global **peak** es igual a **MAX-1**, caso contrario retornará **0**.
- d) (5 ptos) El procedimiento `void push(int stack[], int num)` que agrega un elemento a la pila de la siguiente forma: incrementa en uno la variable global **peak** y luego copia el contenido de **num** en la posición **peak** al arreglo **stack**. Todo lo anterior se realiza SI Y SOLO SI **peak** es menor que la constante **MAX**.
- e) (5 ptos) La función `int pop(int stack[])`, que remueve un elemento de la pila de la siguiente forma: obtiene el contenido del arreglo **stack** en la posición de **peak**, luego decrementa **peak** en uno y finalmente retornará el contenido previamente obtenido. Todo lo anterior se realiza SI Y SOLO SI **peak** es mayor o igual que 0, de no serlo debe retornar -1.
- f) (10 ptos) El procedimiento `void reverse(int stack[], int data[], int n)`, que permite invertir el orden de los **n** números del arreglo **data**, es decir, los números que están en las últimas posiciones pasarán a ser primeros. Use las funciones de la pila antes descritas para invertir el orden de los números en el arreglo **data**.
- g) (10 ptos) Un programa principal que solicite al usuario **n** números a ser ingresados y colocados en un arreglo, el cual será invertido usando el procedimiento *f)* y su contenido mostrado por pantalla. Defina la constante **MAX** y la variable global **peak**.