Introducción a la Programación

Funciones

Edwin Salvador

2 de agosto de 2016

Clase 16

Contenido I

Metodología top-down

- Los pasos para preparar una taza de café:
  1. Ir a la cocina
  2. Calentar agua
  3. Preparar la taza
  4. Poner agua en la taza
- Cada uno de estos pasos implican otros pasos más detallados. Ej. "Ir a la cocina" implica "caminar", "caminar" implica "dar una paso tras otro", etc.
- **Top-down** propone pensar en la solución desde acciones generales y luego ir viendo los detalles de cada una.

Contenido I

Módulos o subprogramas

- Cada una de las acciones que ejecutamos. Ej: “caminar”, “calentar el agua”, etc.
- Cada uno de estos módulos será desarrollados a detalle a medida que se realiza el algoritmo.

Contenido I

Funciones

- Los módulos en la programación se implementan como funciones.
- Las funciones pueden ser invocadas desde el programa principal de desde otras funciones.
- Las funciones pueden realizar alguna tarea y pueden o no retornar algún valor.
- Funciones de biblioteca: `printf`, `scanf`, `strcpy`

Archivo	Descripción	Algunas de las funciones que define
`stdio.h`	Entrada y salida estándar	`printf`, `scanf`, `sprintf`, `getc`, `putc`, `fopen`, `fclose`, `fscanf`, etc.
`string.h`	Utilidades para manejo de cadenas de caracteres	`strcpy`, `strcat`, `ctrcmp`, etc.
`math.h`	Funciones matemáticas	`sin`, `cos`, `log`, `abs`, `pow`, etc.

Figura: Funciones de biblioteca en C

- Las invocamos mediante la línea `#include <stdio.h>` en nuestros programas.

Contenido I

Contenido I

Funciones definidas por el programador

- Las funciones realizan alguna acción en base a los **parámetros** que se le pasen y luego devuelven un valor (**valor de retorno**).
- **Cabecera** parámetros, tipo de valor de retorno y nombre.
- **Cuerpo** la códificación del algoritmo.
- **Prototipo de una función** describe la lista de argumentos, el tipo de dato a retornar y el nombre. Ej:

```
double valorAbsoluto(double);
long unificarFecha(int, int, int);
```

Contenido I

Definición de una función

```
double valorAbsoluto(double d)
{
  double ret = d;
  if( ret<0 )
  {
    ret = -ret;
  }
  return ret;
}
```

La misma función también se la puede implementar así:

```
double valorAbsoluto(double d)
{
  return d<0?-d:d;
}
```

Contenido I

Invocación de la función

```c
#include<stdio.h>
// prototipo de la funcion
double valorAbsoluto(double);

// programa principal
int main()
{
  double v, a;
  printf("Ingrese un valor numerico: "); scanf("%lf",&v);
  // invoco a la funcion
  a = valorAbsoluto(v);
  printf("El valor absoluto de %lf es %lf\n",v,a);
  return 0;
}

// desarrollo de la funcion
double valorAbsoluto(double d)
{
  return d<0?-d:d;
}
```

Contenido I

Convenciones

- **Nombres Simples** en minúscula. Ej: `sumar`, `procesar`, `abrir`.
- **Nombres Compuestos** la primera palabra en minúscula y la primera letra de las demás palabras en Mayúscula. Ej: `valorAbsoluto`, `obtenerFechaNacimiento`, `procesarValores`

Contenido I

Funciones que no retornan ningún valor (void)

- Tipo de datos void es nulo.

```
void saludar()
{
  printf("Hola !!!\n");
  return; // el return es opcional
}
```

Contenido I

Separación de código

- **Archivos de cabecera** contienen los prototipos de funciones, macros, constantes. Se las incluye en el programa principal con la directiva `include`.
- **Archivos de funciones** permiten que las funciones sean reitulizadas por varios programas.
  - `funciones.h`
  - `funciones.c`
  - `principal.c`

Contenido I

Ejercicio 1

- Leer seis valores numéricos enteros. Los primeros 3 representan el día, el mes y el año de una fecha, los tres restantes representan los mismos atributos de otra. Se pide determinar e informar cuál de las dos fechas es posterior.

Soluciones??
Para resolver este problema se utilizará el algoritmo de unificación de fecha.
año * 10000 + mes * 100 + día
2003 * 10000 + 12 * 100 + 5 = 20031205

Algoritmo

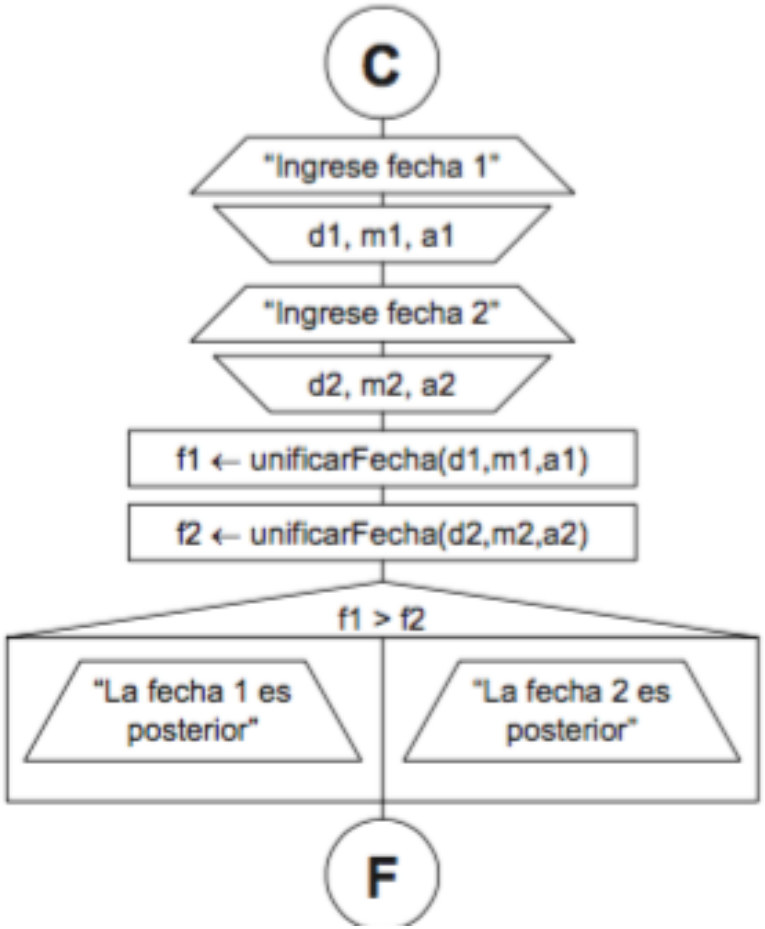
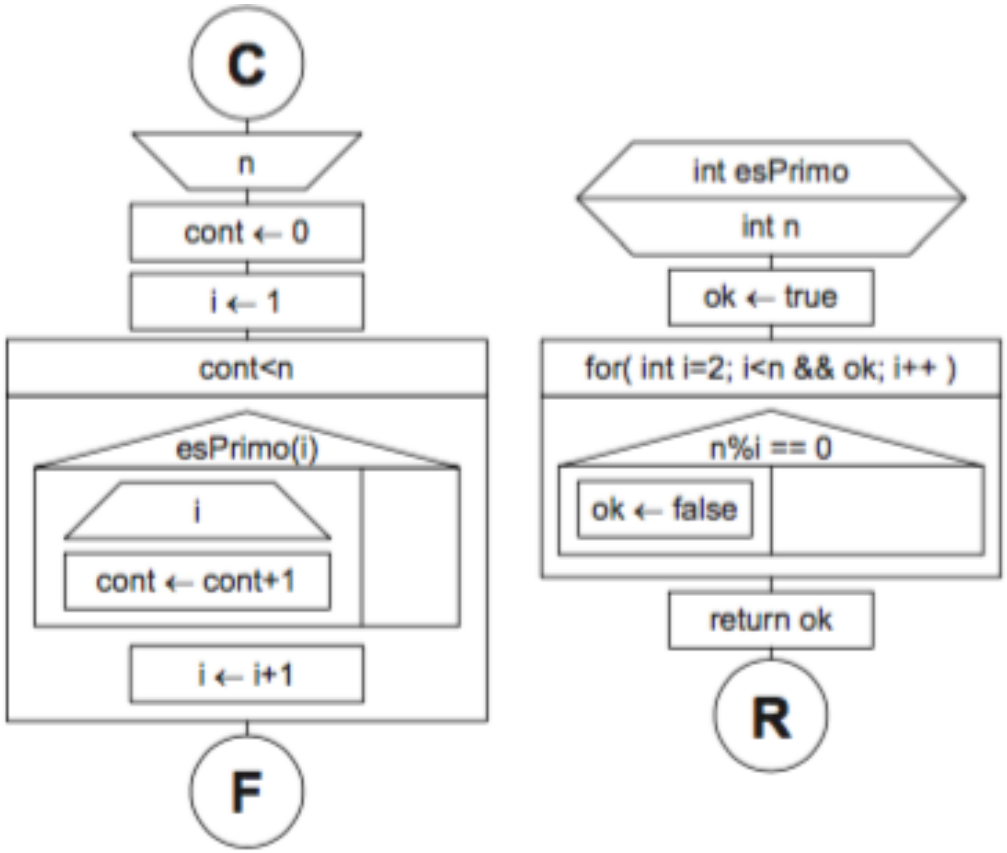


Figura: Qué fecha es posterior

Ejercicio 2

Mostrar los primeros n números primos siendo n un valor que ingresará el usuario.
La sección del programa que determina si el numero es o no primo debe ser desarrollada dentro de una función.



Contenido I

Deber - Ejercicio 1

Se tiene una tabla o planilla con los resultados de la última llamada a examen de una materia, con la siguiente información: matrícula (valor numérico entero de 8 dígitos) nota (valor numérico entero de 2 dígitos entre 1 y 10) nombre (valor alfanumérico de 10 caracteres)
Se pide informar: Nota promedio
Para indicar el fin del ingreso de datos el operador ingresará un registro nulo con matrícula=0, nota=0 y nombre = "".
Se debe implementar una función que pida el ingreso de los datos y otra función que calcule el promedio de las notas.

Deber - Ejercicio 2

Se ingresa por consola un número entero que representa un sueldo que se debe pagar. Considerando que existen billetes de las denominaciones que se indican más abajo; informar, que cantidad de billetes de cada denominación se deberá utilizar, dando prioridad a los de valor nominal más alto. Denominaciones ($) = 100, 50, 20, 10, 5, 2, 1
Ejemplo: sueldo: 5217

Cantidad	Denominación
52	100
0	50
0	20
1	10
1	5
1	2
0	1