



*Universidad Autónoma
de Baja California*



Facultad de Ingeniería Arquitectura y Diseño

*Prácticas de Programación
Estructurada*

Clave: 12125

Pedro Núñez Yépiz

Ensenada Baja California, Agosto 2013

PRACTICA 0

1. Instalacion del IDE (Entorno de Desarrollo Integrado) paratrabajar en “C”

1.1 Instalacion del IDE DevC++ Windows 7 y anteriores

1.1.1 Para La instalación de DevC++ descargar el programa de la siguiente Liga.

http://sourceforge.net/projects/dev-cpp/files/Binaries/Dev-C%2B%2B%204.9.9.2/devcpp-4.9.9.2_setup.exe/download?use_mirror=iweb

1.1.2 Pasos para instalar DevC++

<http://www.slideshare.net/16134817/cmo-instalar-dev-c>

1.2 Instalacion de IDE DevC++ Windows 8

http://sourceforge.net/projects/wxdsgn/files/wxDev-C%2B%2B/Version%207/wxdevcpp_7.3.1_full_setup.exe/download

Puede descargarlos de:

<http://yepiz.webs.com/descargas.htm>

PRACTICA 1

- 1.- Programa que genere la siguiente salida usando un ciclo for
5,10,15,20,25,30

- 2.- Programa que sirva para generar la tabla de multiplicar de un numero dado.

- 3.- Programa que lea la edad de y sexo de 25 alumnos y desplegar la cantidad de alumnos masculinos y la cantidad de femeninos.

NOTA: ESTA PRACTICA ES DE SONDEO.

Enviar practica al correo productos.programacion@gmail.com al termino de la practica, no se aceptaran practicas despues de 30 minutos despues de terminado el horario del laboratorio.

Poner como asunto tu nombre y practica1, usar cuenta oficial de UABC

PRACTICA 2

- 1.- Programa que lea 3 números enteros, Si hay 2 o mas números repetidos desplegar el mensaje de "error", si los 3 numeros son diferentes desplegar los 3 números en orden ascendente.
- 2.- Programa que lea 7 números enteros y desplegar cual de los 7 números es el menor y cual el mayor.
- 3.- Programa que lea 3 calificaciones calcule el promedio del alumno y desplegar:

Si prom < 30 Repetir
Si prom >=30 y prom <60 extraordinario
Si prom >=60 y prom <70 suficiente
Si prom >=70 y prom <80 Regular
Si prom >=80 y prom <90 bien
Si prom >=90 y prom <98 muy bien
Si prom >=98 y prom <=100 excelente
Si prom >100 Error en promedio

Realizar programa en forma de árbol y optimizarlo lo mas posible.

Nota: No usar (&& and) ni (|| or)

Enviar practica al correo productos.programacion@gmail.com al termino de la practica, no se aceptaran practicas despues de 30 minutos despues de terminado el horario del laboratorio.

Poner como asunto tu nombre y practica 2, usar cuenta oficial de UABC

PRACTICA 3

1.- Programa que imprima el factorial de un número dado. (ojo no mayor de 10) la salida deberá ser de la siguiente manera.

Factorial de 5

$1 * 2 = 2$

$2 * 3 = 6$

$6 * 4 = 24$

$24 * 5 = 120$

El factorial de 5 es 120

2.- Programa que Imprima la siguiente salida

1, 2, 3, 2, 3, 4, 3, 4, 5

3.- programa que genere aleatoriamente 50 numero entre el 50 y 100, desplegar la suma y el promedio de los 50 números.

4.- Programa que se repita N cantidad de veces hasta que el usuario presione la tecla ESC. El programa deberá pedir un numero al usuario entre el 1 y 50 realizar y desplegar la suma de todos los números, el promedio y cuantos números son pares e impares

Enviar practica al correo productos.programacion@gmail.com al termino de la practica, no se aceptaran practicas despues de 30 minutos despues de terminado el hoario del laboratorio.

Poner como asunto tu nombre y practica 3, usar cuenta oficial de UABC

PRACTICA 4

1.- Programa que genere n cantidad de números entre el 10 y 50, desplegarlos en pantalla todos.

El programa se deberá detener cuando se generen los primeros 12 números pares. Imprimir la suma y promedio de los números impares

2.- Programa que lea la edad de 10 alumnos (validar edad entre 10 y 70 años) Desplegar la edad del alumno mayor y edad del alumno menor, así como el promedio de las edades.

3.- Programa que simule un reloj digital, (HH:MM:SS) se deberá salir cuando el usuario presione la tecla ESC (**Nota: No es importante la exactitud de los segundos, no usar el tiempo del sistema**)

Enviar practica al correo productos.programacion@gmail.com al termino de la practica, no se aceptaran practicas despues de 30 minutos despues de terminado el hoario del laboratorio.

Poner como asunto tu nombre y practica 4, usar cuenta oficial de UABC

PRACTICA 5

1.- Programa que realice lo siguiente:

- a) Leer una cadena de máximo 40 caracteres.
- b) Con la cadena original generar una nueva cadena con todos los caracteres en mayúsculas
- c) Con la cadena original generar una nueva cadena con todos los caracteres en minúsculas
- d) Con la cadena original generar una nueva cadena, eliminar todos los espacios (solo caracteres alfanuméricos)
- e) Con la cadena sin espacios (del punto d) generar una nueva cadena pero inversa.
- f) Imprimir todas las cadenas**
- g) Imprimir la cantidad de caracteres que tiene la frase
- h) Imprimir la cantidad de cada vocal de la cadena original

- a) Hola Mundo cruel y despiadado
- b) HOLA MUNDO CRUEL Y DESPIADADO
- c) hola mundo cruel y despiadado
- d) HolaMundocruelydespiadado
- e) Odadaipsedyleurcodnumaloh
- g) TU FRASE TIENE 20 caracteres
- h) A=3 E=2 I=1 O=3 U=2

El programa deberá repetirse N cantidad de veces mientras el usuario introduzca una frase y salirse al introducir la frase **NO MAS FRASES** solo en mayúsculas.

NOTA: usar funciones.

Enviar practica al correo productos.programacion@gmail.com al termino de la practica, no se aceptaran practicas despues de 30 minutos despues de terminado el horario del laboratorio.
Poner como asunto tu nombre y practica 5, usar cuenta oficial de UABC

PRACTICA 6

1.- Programa que llene una matriz de 4 x 4 con números generados aleatoriamente entre el 1 y 16 (no repetidos) desplegar los datos de la matriz

```
      M A T R I Z
5     7     1     15
12    11     2     14
9     16     8     4
3     10    13     6
```

2.- Programa que genere un vector de **20** con números aleatorios entre el 50 y 80 (no repetidos). Mostrar todo los datos del vector, el programa se deberá repetir hasta que el usuario lo desee.

Nota: Usar funciones, no se permiten variables globales

EJEMPLO DE ENVIO DE ARREGLOS COMO PARÁMETROS:

Prototipo:

```
void funcion ( int v[16 ]);
```

Desarrollo de la función:

```
void funcion (int v[16 ])  
{  
    int i;  
    for(...  
        v[i]=rand%50; //se llena el vector y regresa con los  
                                // nuevos valores de donde fue  
                                //invocada la funcion  
    }  
}
```

Llamada a la función:

```
void principal ()  
{  
    Int vector[20];  
    funcion(vector); //Regresa con los nuevos datos....  
    for(...  
        printf("[%2d] -> %d",i,vector[i]);  
    }  
}
```

Salida a pantalla:

```
[ 0]-> 55  
[ 1]->78  
...  
[19]->62
```

Enviar practica al correo productos.programacion@gmail.com al termino de la practica, no se aceptaran practicas despues de 30 minutos despues de terminado el horario del laboratorio. Poner como asunto tu nombre y practica 6, usar cuenta oficial de UABC

PRACTICA 7

1.- Programa que contenga el siguiente menu

MENU

- 1.- Agregar
- 2.- Mostra Todo
- 3.- Salir

El programa deberá poder almacenar en un arreglo (**máximo 30 registros**) los datos básicos de un alumno (**matricula, nombre, edad, sexo**).

NOTA: Usar **typedef**, y **struct** para almacenar los datos utilizarlo en el arreglo.

NOTA 2: Usar funciones, no se permiten variables globales

Enviar practica al correo productos.programacion@gmail.com al termino de la practica, no se aceptaran practicas despues de 30 minutos despues de terminado el horario del laboratorio.
Poner como asunto tu nombre y practica 6, usar cuenta oficial de UABC

EJEMPLO Y TEORIA BASICA DE typedef

Definición de tipos: typedef

Se puede dar un nombre nuevo a cualquier tipo de datos mediante **typedef**.
La sintaxis es

typedef declaración;

donde **declaración** tiene la forma de una declaración de variable, sólo que se está definiendo un tipo de datos.

```
typedef long pareja [2];
```

define un tipo **pareja** que se puede usar en declaraciones de variables:

```
pareja p;
```

es equivalente a

```
long p [2];
```

Ejemplos de typedef con estructuras

```
typedef struct Persona PERSONA;
PERSONA dato; /* igual que struct Persona dato; */
```

Un uso típico es la redefinición de tipos estructurados:

```
typedef struct      /* estructura anónima */
{
    char nombre[80];
    char sexo;
    int edad;
} Persona;          /* se declara el tipo Persona */

...
Persona p;         // Declaracion de la variable p del tipo de dato Persona
...
p.edad = 44;
```

PRACTICA 8

1.- Programa que contenga el siguiente menu

MENU

- 1.- Agregar
- 2.- Eliminar
- 3.- Buscar Registro
- 4.- Mostra Todo
- 5.- Salir

El programa deberá poder almacenar los datos en un **archivo binario** los datos básicos de un alumno (**matricula, nombre, edad, sexo**).

NOTA: Usar funciones, no se permiten variables globales

En esta liga pues encontrar todo lo referente a archivos y ejemplos

<http://c.conclase.net/ficheros/index.php?cap=001#inicio>

Enviar practica al correo productos.programacion@gmail.com al termino de la practica, no se aceptaran practicas despues de 30 minutos despues de terminado el horario del laboratorio.
Poner como asunto tu nombre y practica 6, usar cuenta oficial de UABC

Anexo 2: MODO GRAFICO ALLEGRO V 5.

Referencia Oficial:

<http://alleg.sourceforge.net/>

Manual de referencia

<https://www.allegro.cc/manual/5/index.html>

Manual Allegro en español

http://artemisa.unicauca.edu.co/~wpantoja/web/graficos_gcc/allegro/manual_allegro.pdf

BIBLIOGRAFIA:

COMO PROGRAMAR EN C/C++, Deitel & Deitel, Prentice Hall,
ISBN:9688804711

C/C++ CURSO DE PROGRAMACIÓN, Fco. Javier Ceballos, Alpha
Omega ISBN:978847897628

FUNDAMENTOS DE PROGRAMACIÓN, Jose Luis Joyanes A, ed.
McGrawhill, ISBN:9788448139865

<http://alleg.sourceforge.net/>

<http://yepiz.webs.com>