UNIVERSIDAD DE MÁLAGA
DPTO. DE LENGUAJES Y CC. DE LA COMPUTACIÓN
E.T.S. DE INGENIERÍA INFORMÁTICA

SOLUCIÓN
A
Práctica Nº 1.
Entorno de Programación Code::Blocks e Introducción a C++

Preliminares.

1. El entorno de programación Code::Blocks se introducirá mediante la resolución de un determinado problema bajo la dirección del profesor.

Ejercicios de clase.

2. El siguiente programa escrito en C++ calcula la cantidad bruta y neta a pagar por un trabajo realizado en función de las horas y días trabajados. Sin embargo, en el momento en que se intenta compilarlo se producen una serie de errores. El alumno debe localizar dichos errores y corregirlos. Para ello debe examinar los mensajes que proporciona el compilador e interpretarlos convenientemente.

```
#include <iostream>
using namespace std;

const tasa : 25.0;
const PRECIO_HORA = 60.0;

int main()
{
      double horas,dias,total,neto;

      cout << "Introduzca las horas trabajadas: ";
      cin << horas;
      cout << "Introduzca los dias trabajados: ";
      cin >> dias;
      horas*dias*PRECIO_HORA = total;
      neto = total-TASA;
      cout << "El valor total a pagar es: ";
      cout << total << endl;
      cout << "El valor neto a pagar es: ";
      cout << NETO << endl;

      return 0;
}
```

SOLUCIÓN:

```
#include <iostream>

using namespace std;

//const tasa : 25.0;
  const double tasa = 25.0;
//const precio_hora = 60.0;
  const double precio_hora = 60.0;
int main()
{
	double horas,dias,total,neto;

	cout << "Introduzca las horas trabajadas: ";
	//cin << horas;
	  cin >> horas;
	cout << "Introduzca los dias trabajados: ";
	cin >> dias;
	//horas*dias*precio_hora = total;
	  total = horas*dias*precio_hora;
	//neto = total-Tasa;
	  neto = total-tasa;
	cout << "El valor total a pagar es: ";
	cout << total << endl;
	cout << "El valor neto a pagar es: ";
	//cout << NETO << endl;
	  cout << neto << endl;

	return 0;
}
```

3. Escriba un programa que acepte un dato de tipo **int** de teclado y posteriormente lo escriba en pantalla. Ejecútelo introduciendo un número **int** válido, y posteriormente ejecútelo introduciendo por teclado un dato que no pertenezca al tipo **int**, por ejemplo una palabra cualquiera. Añada un comentario al principio del programa en el que explique cuáles son las diferencias que ha encontrado entre ambas ejecuciones del mismo programa.

SOLUCIÓN:

```
/*
* Cuando se introduce un número entero válido, la salida es el mismo número.
* Cuando no se introduce un número entero válido, la salida es un número
* entero que nada tiene que ver con la entrada, por ejemplo sale un 0
*/

#include <iostream>
using namespace std;


int main()
{
	int numero;

	cout << "Introduzca un numero entero: ";
	cin >> numero;
	cout << "El numero introducido es: " << numero << endl;
```

```
	return 0;
}
```

4. Escriba un programa que visualice por pantalla el tamaño en bytes que ocupan todos y cada uno de los tipos básicos vistos en clase: **int**, **long**, **char**, etc. Para ello debe usarse el operador **sizeof**.

SOLUCIÓN:

```
#include <iostream>
using namespace std;

int main() {

	cout << "bool: " << sizeof(bool) << endl;
	cout << "char: " << sizeof(char) << endl;
	cout << "short: " << sizeof(short) << endl;
	cout << "unsigned short: " << sizeof(unsigned short) <<
	endl; cout << "int: " << sizeof(int) << endl;
	cout << "unsigned: " << sizeof(unsigned) << endl;
	cout << "long: " << sizeof(long) << endl;
	cout << "unsigned long: " << sizeof(unsigned long) << endl;
	cout << "float: " << sizeof(float) << endl;
	cout << "double: " << sizeof(double) << endl;
	cout << "long double: " << sizeof(long double) << endl;

	return 0;

}
```

5. Escriba un programa que lea cuatro letras por teclado, y posteriormente escriba dichas letras de manera que cada una de ellas se encuentre codificada sustituyéndola por aquel carácter que le sigue en la tabla de código ASCII. Por ejemplo, ante la entrada **SETO** debe producir la salida **TFUP**.

SOLUCIÓN:

```
#include <iostream>


using namespace std;

// version 1
int main()
{
	char c1,c2,c3,c4;

	cout << "Introduzca una palabra de cuatro letras: ";
	cin >> c1 >> c2 >> c3 >> c4;
	c1 = char(int(c1)+1);// o bien c1 = char(c1+1); o bien c1 = c1+1;
	c2 = char(int(c2)+1); c3 =
	char(int(c3)+1);    c4    =
	char(int(c4)+1);
```

```
	cout << "El resultado de su codificacion es: ";
	cout << c1 << c2 << c3 << c4 << endl;

	return 0;
}
/*
// version 2
int main()
{
	char c;

	cout << "Introduzca una palabra de cuatro letras: ";
	cin >> c;
	cout << "El resultado de su codificacion es: ";
	c = char(int(c)+1);		// o bien c = char(c+1); o bien c = c+1;
	cout << c;
	cin >> c;
	c = char(int(c)+1);
	cout << c;
	cin >> c;
	c = char(int(c)+1);
	cout << c;
	cin >> c;
	c = char(int(c)+1);
	cout << c << endl;


	return 0;

}
*/
```

6. Escriba un programa que lea de teclado un número natural, que representa una cierta cantidad de Bytes, y muestre por pantalla los MBytes, KBytes y Bytes que podemos obtener. Por ejemplo, dado el número 26871979, el resultado sería 25 MBytes, 642 KBytes y 171 Bytes, ya que 26871979 Bytes = 25 MBytes + 642 KBytes + 171 Bytes.

SOLUCIÓN:

```
#include <iostream>

using namespace std;


const int BYTES_EN_KBYTE = 1024;
const int KBYTES_EN_MBYTE = 1024;
const int BYTES_EN_MBYTE = BYTES_EN_KBYTE * KBYTES_EN_MBYTE;


int main()
{
	int bytes;

	cout << "Introduzca una cantidad de Bytes: ";
	cin >> bytes;
	cout << "Esa cantidad forma:" << endl;
	cout << "	Mbytes = " <<  bytes / BYTES_EN_MBYTE << endl;
```

```
        bytes = bytes % BYTES_EN_MBYTE;
        cout << "     Kbytes = " <<  bytes / BYTES_EN_KBYTE << endl;
        bytes = bytes % BYTES_EN_KBYTE;
        cout << "     Bytes = " <<  bytes << endl;

        return 0;
}
```

Ejercicios de refuerzo.

7. El siguiente programa escrito en C++ calcula el área y la longitud de una circunferencia. Al compilarlo se producen una serie de errores que el alumno debe localizar y corregir.

```
#include <iostream>
using namespace std;

const int PI=3.1416

int main()
{
  double longitud ,area;
  int radio;

  out << "Hola" ; << endl;
  cout<< "Este programa calcula la longitud y el área de un círculo"

  cin << "Radio = " >> radio;

  long = 2*PI*radio;
  area = PI*(radio*radio)
  cout << 'area = '<< area << endl;
  cout << 'long = '<< area << endl;

  return 0;         // Valor de retorno al S.O.
}
```

SOLUCIÓN:

```
#include <iostream>
using namespace std;

//const int PI=3.1416
const double PI=3.1416;

int main()
{
  double longitud ,area;
  int radio;

  //out << "Hola" ; << endl;
  cout << "Hola" << endl;

  //cout<< "Este programa calcula la longitud y el área de un círculo";
  cout<< "Este programa calcula la longitud y el área de un círculo\n";

  //cin << "Radio = " >> radio;
  cout  << "Radio = ";
  cin >> radio;
```

```
  //long = 2*PI*radio;
  longitud = 2*PI*radio;
  //area = PI*(radio*radio)
  area = PI*(radio*radio);
  //cout << 'area = '<< area << endl;
  cout << "area = " << area << endl;
  //cout << 'long = '<< area << endl;
  cout << "longitud = " << longitud << endl;

  return 0;     // Valor de retorno al S.O.
}
```

8. Escriba un programa que lea una palabra de cuatro letras minúsculas por teclado y a continuación la escriba en mayúsculas.

SOLUCIÓN:

```
#include <iostream>

using namespace std;

// version 1
int main()
{
      char c1,c2,c3,c4;

      cout << "Introduzca una palabra de cuatro letras minusculas: ";
      cin >> c1 >> c2 >> c3 >> c4;
      c1 = char(int('A')+(int(c1)-int('a')));
      // o bien c1 = 'A'+(c1-'a');
      c2 = char(int('A')+(int(c2)-int('a')));
      c3 = char(int('A')+(int(c3)-int('a')));
      c4 = char(int('A')+(int(c4)-int('a')));
      cout << "La palabra en mayusculas es: ";
      cout << c1 << c2 << c3 << c4 << endl;

      return 0;
}
/*
// version 2
int main()
{
      char c;

      cout << "Introduzca una palabra de cuatro letras minusculas: ";
      cin >> c;
      cout << "La palabra en mayusculas es: ";
      c = char(int('A')+(int(c)-int('a')));
      // o bien c = 'A'+(c-'a');
      cout << c;
      cin >> c;
      c = char(int('A')+(int(c)-int('a')));
      cout << c;
      cin >> c;
      c = char(int('A')+(int(c)-int('a')));
      cout << c;
      cin >> c;
      c = char(int('A')+(int(c)-int('a')));
```

```
	cout << c << endl;

	return 0;
}
*/
```

9. Escriba un programa que sólo declare variables de tipo **int**. El programa deberá leer dos números desde el teclado; posteriormente los sumará y almacenará el resultado en una variable; finalmente escribirá por pantalla el resultado de la suma. Ejecute el programa con datos cualesquiera y verifique que funciona. Después ejecute dicho programa tomando como datos de entrada 1 y 3000000000. ¿Por qué no funciona?

SOLUCIÓN:

```
// Porque 3000000000 excede el valor máximo permitido para los enteros


#include <iostream>

using namespace std;


int main()
{
	int n1,n2,suma;

	cout << "Introduzca dos numeros enteros: ";
	cin >> n1 >> n2;
	suma = n1 + n2;
	cout << "Su suma es: " << suma << endl;

	return 0;
}
```

10. Escriba un programa que lea por teclado una cierta cantidad de segundos y muestre su equivalente en semanas, días, horas, minutos y segundos. Por ejemplo, si se lee la cantidad de 3672 segundos, la salida será: 0 semanas, 0 días, 1 hora, 1 minuto, 12 segundos.

SOLUCIÓN:

```
#include <iostream>

using namespace std;


const int SEG_EN_MIN = 60;
const int MIN_EN_HOR = 60;
const int SEG_EN_HOR = SEG_EN_MIN * MIN_EN_HOR;
const int HOR_EN_DIA = 24;
const int SEG_EN_DIA = SEG_EN_HOR * HOR_EN_DIA;
const int DIA_EN_SEM = 7;
```

```
const int SEG_EN_SEM = SEG_EN_DIA * DIA_EN_SEM; int
main()
{
	unsigned segundos;

	cout << "Introduzca una cantidad de segundos: ";
	cin >> segundos;
	cout << "Esa cantidad forma:" << endl;
	cout << "	semanas = " <<  segundos / SEG_EN_SEM << endl;
	segundos = segundos % SEG_EN_SEM;
	cout << "	dias = " <<  segundos / SEG_EN_DIA << endl;
	segundos = segundos % SEG_EN_DIA;
	cout << "	horas = " <<  segundos / SEG_EN_HOR << endl;
	segundos = segundos % SEG_EN_HOR;
	cout << "	minutos = " <<  segundos / SEG_EN_MIN << endl;
	segundos = segundos % SEG_EN_MIN;
	cout << "	segundos = " <<  segundos << endl;

	return 0;
}
```

11. Escriba un programa que calcule la nota final de una asignatura. Para ello deberá leer por teclado la nota de la parte de teoría y la nota de la parte de problemas, y habrá de calcular la nota final considerando que la parte de teoría vale un 70% de la nota final y la de práctica un 30%.

SOLUCIÓN:

```
#include <iostream>
using namespace std;

const double PORCENTAJE_TEORIA = 0.7;
const double PORCENTAJE_PRACTICA = 0.3;


int main() {

	double teoria, practica, calificacion;

	cout << "Nota Teoria: ";
	cin >> teoria;
	cout << "Nota Practica: ";
	cin >> practica;
	calificacion = teoria * PORCENTAJE_TEORIA +
				practica * PORCENTAJE_PRACTICA;
	cout << "La calificacion es: " << calificacion << endl;

	return 0;

}
```

12. Escriba el siguiente código, ejecútelo y descubra qué hace este programa y cómo lo hace.

```
#include <iostream>
using namespace std;

int main()
{
  int a=6, b=14;
  int auxiliar;
  cout << "a vale " << a << " y b vale " << b << endl;

  // ¿Qué hacen estas tres sentencias?
  auxiliar = a;
  a = b;
  b = auxiliar;

  cout << "a vale " << a << " y b vale " << b << endl;

  return 0;
}
```

Sustituya las tres asignaciones que hay tras el comentario por estas otras tres:

```
        a = a + b;
        b = a - b;
        a = a - b;
```

Puede comprobarse que el resultado es análogo al caso anterior: estudie cómo funciona este nuevo programa.

SOLUCIÓN:

```
/*
* Tanto uno como otro grupo de sentencias llevan a cabo el intercambio de
* los valores almacenados en dos variables (a y b). En el primer caso se
* hace uso de una variable intermedia (auxiliar)
*/
```