

# CS1102 Programación Orientada a Objetos 1

## Unidad 1: Estructuras de control

### Asesoría - Semana 3

#### Asesores:

Plinio Avendaño

[plinio.avendano@utec.edu.pe](mailto:plinio.avendano@utec.edu.pe)

Alonso Barrios

[alonso.barrios@utec.edu.pe](mailto:alonso.barrios@utec.edu.pe)

Alex Loja

[alex.loja@utec.edu.pe](mailto:alex.loja@utec.edu.pe)

Sebastian Peñaranda

[sebastian.penaranda@utec.edu.pe](mailto:sebastian.penaranda@utec.edu.pe)



**Autores:**

**María Hilda Bermejo**

# 1

## Estructuras de Control



# Ejercicio 1:

Elena es una profesora de UTEC, ella desea implementar un programa que le indique la cantidad de alumnos aprobados y el promedio de nota de sus alumnos.

**Nota:** Para aprobar un curso la nota debe ser mayor igual a 10.5.



**Input:**

{7, 2, 10.5, 12, 18, 20, 12.9, 13.1}



**Output:**

Aprobados: 6

Promedio: 11.9375



**Input:**

{15, 18, 7, 11, 8.5}



**Output:**

Aprobados: 3

Promedio: 11.9

# Solución:

```
1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5
6  const double APPROVE_GRADE = 10.5;
7
8  int main(int argc, char *argv[]) {
9      vector<double> grades = {7, 2, 10.5, 12, 18, 20, 12.9, 13.1};
10
11     double average = 0;
12     int approved = 0;
13     int count = 0;
14     double sum = 0;
15
16     for(auto grade : grades) {
17         if(grade ≥ APPROVE_GRADE) {
18             approved++;
19         }
20
21         sum += grade;
22         count++;
23     }
24
25     average = sum / count;
26
27     cout << "Aprobados: " << approved << endl;
28     cout << "Promedio: " << average << endl;
29
30     return 0;
31 }
```

## Ejercicio 2:

Escribe un programa que solicite una contraseña, y que vuelva a solicitarla hasta que coincidan. El usuario tiene hasta un máximo de 3 oportunidades para hacer coincidir la contraseña.

**Nota:** La contraseña no puede tener espacios en blanco.



## Solución:

```
#include <iostream>
#include <string>
#include <vector>

using namespace std;

int main(int argc, char *argv[]) {
    string password = "";
    string confirm_password = "";
    int attempts = 3;

    cout << "Ingrese su contraseña: ";
    getline(cin, password, '\n');

    while(attempts > 0) {
        cout << "Confirme su contraseña: ";
        getline(cin, confirm_password, '\n');

        if(password == confirm_password) {
            cout << "¡Contraseña confirmada!" << endl;
            break;
        }

        attempts--;
        cout << "¡Contraseña incorrecta! Le quedan " << attempts << " intentos" << endl;
    }

    return 0;
}
```

# Ejercicio 3:

Convierta todas las vocales de una cadena de texto en mayúsculas, y cambie todos los espacios por un *underscore* ( \_ ).



**Input:**

Hola como te encuentras



**Output:**

HOIA\_cOmO\_tE\_EncUEntrAs



# Solución:

```
#include <iostream>

using namespace std;

int main(int argc, char *argv[]) {
    string text = "Hola como te encuentras";

    for(int i = 0; i < text.size(); ++i) {
        switch (text[i]) {
            case 'a':
            case 'e':
            case 'i':
            case 'o':
            case 'u':
                text[i] = toupper(text[i]);
                break;
            case ' ':
                text[i] = '_';
            default: break;
        }
    }

    cout << text << endl;

    return 0;
}
```

# 2

## Funciones



# Ejercicio 1:

Dada una matriz de productos, haga una función que reciba como parámetro el *id* del producto. La función debe retornar los datos del producto seleccionado.

**Nota:** La matriz tendrá la siguiente estructura: **id | nombre | precio | procedencia**

## Matriz:

```
{{"H102", "Leche", "2.50", "Peru"},  
{"H103", "Pescado", "3.00", "Chile"},  
{"P122", "Huevo", "5.50", "Bolivia"},  
{"A202", "Papa", "4.70", "Brasil"},  
{"F102", "Aceite", "12.30", "Espana"}}
```



**Input:**

H103



**Output:**

Pescado

3.00

Chile

## Solución:

```
#include <iostream>

using namespace std;

const int row = 5;
const int column = 4;

void find(const string& id, string matrix[][column]) {
    for(int i = 0; i < row; i++) {
        if(matrix[i][0] == id) {
            for(int j = 1; j < column; j++) {
                cout << matrix[i][j] << endl;
            }

            return;
        }
    }

    cout << "¡Producto no encontrado!\n";
}

int main(int argc, char *argv[]) {
    string products[row][column] = {
        {"H102", "Leche", "2.50", "Peru"},
        {"H103", "Pescado", "3.00", "Chile"},
        {"P122", "Huevo", "5.50", "Bolivia"},
        {"A202", "Papa", "4.70", "Brasil"},
        {"F102", "Aceite", "12.30", "Espana"}
    };

    find("H103", products);
    return 0;
}
```

## Ejercicio 2:

Crea un programa que calcule tu promedio en determinado curso.

**Ejemplo:** Economías en Desarrollo: 30% EF + 30% PC + 40% TG



**Input:**

EF: 12

PC: 10

TG: 14



**Output:**

12.2



# Solución:

```
#include <iostream>
#include <vector>
#include <math.h>

using namespace std;

double calculate_average(vector<double> grades, vector<double> percentage) {
    double average = 0;

    for(int i = 0; i < percentage.size(); i++) {
        average += grades[i] * percentage[i];
    }

    return average;
}

void get_grades(double& ef, double& pc, double& tg) {
    cout << "EF: ";
    cin >> ef;

    cout << "PC: ";
    cin >> pc;

    cout << "TG: ";
    cin >> tg;
}

int main(int argc, char *argv[]) {
    vector<double> percentage = {0.3, 0.3, 0.4};
    vector<double> grades;

    double ef = 0;
    double pc = 0;
    double tg = 0;

    get_grades(ef, pc, tg);
    grades.push_back(ef);
    grades.push_back(pc);
    grades.push_back(tg);

    double average = calculate_average(grades, percentage);
    cout << average << endl;

    return 0;
}
```

## Ejercicio 3:

Escribir un programa que imprima los números de 1 a  $n$ , sustituyendo los múltiplos de 3 por “Fizz”, los múltiplos de 5 por “Buzz” y los múltiplos de 3 y 5 por “FizzBuzz”.

**Nota:** El valor de  $n$  es ingresado por el usuario.



**Input:**

15



**Output:**

1 2 Fizz 4 Buzz Fizz 7 8 Fizz Buzz  
11 Fizz 13 14 FizzBuzz

# Solución:

```
#include <iostream>

using namespace std;

void fizzbuzz(int number) {
    for(int i = 1; i <= number; i++) {
        if(i % 3 == 0 && i % 5 == 0) {
            cout << "FizzBuzz ";
        } else if(i % 3 == 0) {
            cout << "Fizz ";
        } else if(i % 5 == 0) {
            cout << "Buzz ";
        } else {
            cout << i << " ";
        }
    }
}

int main(int argc, char *argv[]) {
    int number = 0;

    cout << "Ingrese un numero: ";
    cin >> number;

    fizzbuzz(number);

    return 0;
}
```

# Repositorio:

<https://github.com/alonso804/POO-I-Asesorias>



# Gracias

Nos vemos en la siguiente asesoría!

