 UNIVERSIDAD MARIANO GÁLVEZ DE GUATEMALA

FACULTAD DE INGENIERÍA EN SISTEMAS DE INFORMACIÓN

**NOMBRE:** Cristian Coronado **CARNET:** 9941-19-4190 **SECCION:** B **FECHA:** 01-03-2020  
**CURSO:** Programación I **CATEDRÁTICO:** Ing.Francisco Aroche

# **HOJA DE TRABAJO No. 3**

/\* **EJERCICIO 1**

El presidente de un club de futbol requiere calcular el sueldo de sus jugadores

si se tiene como dato la edad y la nacionalidad del jugador.

Ademas se sabe que el sueldo se calculara de la siguiente manera: Sueldo fijo $2500.00 Si es extranjero recibe un bono de $500.00

Si la edad esta entre 15 y 20, el salario incrementa en $1400.00

Si la edad esta entre 21 y 25, el salario incrementa en $1500.00

Si la edad esta entre 26 y 30, el salario incrementa en $1200.00

Si la edad es mayor a 30, el salario incrementa en $800.00

Determine el sueldo del jugador si se tiene como datos la edad y nacionalidad.

Mostrar un reporte del total a pagar de planilla, así como el total de jugadores por

los rangos de edad. Ademas de indicar que rango de edad es el mas remunerado.

\*/

#include <fstream>

#include <iostream>

#include <string>

#include <vector>

using namespace std;

//Declaración de estructura de tipo Jugador

struct Jugadores{

string nombre;

string apellido;

int edad;

string esGuatemalteco;

float sueldo;

};

//Creación de vector de tipo estructura de Jugadores

std::vector<Jugadores> jugador;

//Función encargada de agregar jugadores a archivos

void agregarJugadores(){

/\*----DECLARACIÓN DE VARIABLES

\_jugador es de tipo estructura de Jugadores

guatemalteco recibe un valor de tipo string para asignar nacionalidad

\*/

Jugadores \_jugador;

string guatemalteco;

cout << "----------------------------REGISTRO DE JUGADOR----------------------------" << endl;

//Ingreso de nombre

cout << "Ingrese nombre del jugador: " << "\n";

cin >> \_jugador.nombre;

//Ingreso del apellido

cout << "Ingrese apellido del jugador: " << "\n";

cin >> \_jugador.apellido;

//Ingreso de edad

cout << "Ingrese edad del jugador: " << "\n";

cin >> \_jugador.edad;

//Ingreso de nacionalidad

cout << "¿El jugador es de nacionalidad guatemalteca? (Si/No)" << "\n";

cin >> \_jugador.esGuatemalteco;

//Ingreso de Sueldo

cout << "Ingrese el sueldo del jugador: " << "\n";

cin >> \_jugador.sueldo;

//Validación correspondiente a verificar si el jugador ingresado es guatemalteco, si no lo es se asigna el valor de Extranjero

if (\_jugador.esGuatemalteco == "SI" || \_jugador.esGuatemalteco == "si" || \_jugador.esGuatemalteco == "Sí" || \_jugador.esGuatemalteco == "sí"){

\_jugador.esGuatemalteco = "Guatemalteco";

}else{

\_jugador.esGuatemalteco = "Extranjero";

\_jugador.sueldo = \_jugador.sueldo + 500.00;

}

/\*

Validaciones correspondientes a las edades de los jugadores

para determinar el aumento en el salario correspondiente.

\*/

if (\_jugador.edad >= 15 and \_jugador.edad <= 20){

\_jugador.sueldo = \_jugador.sueldo + 1400.00;

}

else if(\_jugador.edad >= 21 and \_jugador.edad <= 25){

\_jugador.sueldo = \_jugador.sueldo + 1500.00;

}

else if(\_jugador.edad >= 26 and \_jugador.edad <= 30){

\_jugador.sueldo = \_jugador.sueldo + 1200.00;

}

else if(\_jugador.edad >30){

\_jugador.sueldo = \_jugador.sueldo + 800.00;

}

//Grabar Jugador en archivo

ofstream archivoJugador("JUGADORES.txt", std::ios\_base::app);

archivoJugador <<

\_jugador.nombre << "\t"<< \_jugador.apellido << "\t" << \_jugador.edad << "\t" <<

\_jugador.esGuatemalteco << "\t" << \_jugador.sueldo

<< endl;

//Cerrar Archivo

archivoJugador.close();

}

//Función encargada de calcular el total a pagar de planilla

void totalPlanilla(){

/\* DECLARACIÓN DE VARIABLES

total variable que almacenará el total a pagar en planilla

\*/

float total = 0.00;

/\*

Iteración de todos los sueldos registrados en el vector "jugador"

para realizar la suma correspondiente al total a pagar en planilla.

\*/

for (int i = 0; i <= jugador.size(); i++){

total = total + jugador[i].sueldo;

}

//Muestra el total a pagar

cout << "-------------------------------------------------------" << "\n";

cout << "EL TOTAL A PAGAR EN PLANILLA ES DE: $" << total << endl;

cout << "-------------------------------------------------------" << "\n";

return;

}

/\*

Función encargada de mostrar el rango de edad más remunerado

y el total de jugadores por edad.

\*/

void edadMasRemunerada(){

//DECLARACIÓN DE VARIABLES

string rangoEdad; //Almacena el rango de edad más remunerado

float suma1520 = 0.00; //Suma de salarios de las edades entre 15 y 20 años

float suma2125 = 0.00; //Suma de salarios de las edades entre 21 y 25 años

float suma2630 = 0.00; //Suma de salarios de las edades entre 26 y 30 años

float sumaMayor30 = 0.00; //Suma de salarios de jugadores mayores a 30 años

int total1520 = 0; //Contador de jugadores de entre 15 y 20 años

int total2125 = 0; //Contador de jugadores de entre 21 y 25 años

int total2630 = 0; //Contador de jugadores de entre 26 y 30 años

int totalMayor30 = 0; //Contador de jugadores mayores a 30 años

float remuneracion = 0.00; //Variable encargada de almacenar el total de remuneración más alta.

/\*

Iteración correspondiente al vector "jugador" para determinar

el total de jugadores y el total de salarios por rango de edad

\*/

for (int i = 0; i <= jugador.size(); i ++){

if(jugador[i].edad >= 15 and jugador[i].edad <= 20){

suma1520 = suma1520 + jugador[i].sueldo;

total1520 = total1520 +1;

}

else if(jugador[i].edad >= 21 and jugador[i].edad <= 25){

suma2125 = suma2125 + jugador[i].sueldo;

total2125 = total2125 + 1;

}

else if(jugador[i].edad >= 26 and jugador[i].edad <= 30){

suma2630 = suma2630 + jugador[i].sueldo;

total2630 = total2630 + 1;

}

else if(jugador[i].edad > 30){

sumaMayor30 = sumaMayor30 + jugador[i].sueldo;

totalMayor30 = totalMayor30 +1;

}

}

//Arreglo encargado de almacenar los totales de salarios según rango de edad.

float arregloTotales[4] = { suma1520, suma2125, suma2630, sumaMayor30 };

/\*

Iteración del arreglo para determinar cuál es la mayor

remuneración de salario y determinar en qué rango de edades.

\*/

for (int i = 0; i <= 3; i++){

if (arregloTotales[i] > remuneracion){

remuneracion = arregloTotales[i];

if (i = 0){

rangoEdad = "15 - 20 AÑOS";

}

else if (i = 1){

rangoEdad = "21 - 25 AÑOS";

}

else if (i = 2){

rangoEdad = "26 - 30 AÑOS";

}

else if (i = 3){

rangoEdad = "MAYOR A30 AÑOS";

}

}

}

//Salida en consola de la función

cout << "-------------------------------------------------------" << "\n";

cout << "EL TOTAL DE JUGADORES DE ENTRE 15 A 20 AÑOS ES " << total1520 << endl;

cout << "EL TOTAL DE JUGADORES DE ENTRE 21 A 25 AÑOS ES " << total2125 << endl;

cout << "EL TOTAL DE JUGADORES DE ENTRE 26 A 30 AÑOS ES " << total2630 << endl;

cout << "EL TOTAL DE JUGADORES MAYORES A 30 AÑOS ES " << totalMayor30 << endl;

cout << "-------------------------------------------------------" << "\n";

cout << "-------------------------------------------------------" << "\n";

cout << "EL RANGO DE EDAD MÁS REMUNERADO ES " << rangoEdad << " CON UN TOTAL DE $" << remuneracion << endl;

cout << "-------------------------------------------------------" << "\n";

return;

}

//Función encargada de extraer datos de archivo

void leerArchivo(){

//Declaración de variable de tipo archivo.

ifstream archivoJugadores;

//Declaración de variable de tipo estructura de Jugadores

Jugadores stJugadores;

//Abrir archivo Jugadores

archivoJugadores.open("JUGADORES.txt", ios::in);

//Obtener los datos desde el archivo.

while(archivoJugadores >> stJugadores.nombre >> stJugadores.apellido >> stJugadores.edad >> stJugadores.esGuatemalteco >> stJugadores.sueldo){

//Guardar los datos obtenidos en un vector

jugador.push\_back(stJugadores);

}

//Cerrar archivo

archivoJugadores.close();

return;

}

//Función principal

int main(){

//Declaración de variables

char opcion; //Guarda la opción ingresada por el usuario

string vacio; //Variable encargada de obtener un valor vacío para selección de opción en menú

bool terminar = false; //Variable bandera para terminar ejecución de programa

//MENÚ DE PROGRAMA

do {

cin.clear();

cout << "---------------------JUGADORES---------------------" << endl;

cout << "1. Agregar jugadores" << "\n"

<< "2. Mostrar total a pagar en planilla" << "\n"

<< "3. Rango de edad más remunerado y total de jugadores por edad" << "\n"

<< "4. Salir" << endl;

cout << "Elija una opción (1-4)" << endl;

cin >> opcion;

switch(opcion){

case '1':

agregarJugadores();

break;

case '2':

leerArchivo();

totalPlanilla();

break;

case '3':

leerArchivo();

edadMasRemunerada();

break;

case '4':

terminar = true;

break;

default:

cout << "Opción inválida";

getchar();

getline(cin, vacio);

break;

}

}while(terminar != true);

return 0;

}

**EJERCICIO 2**

/\*

En la Pizzeria "Gran sabor" se tienen las siguientes ofertas:

Por la compra de una pizza grande (Q80.00), recibe un descuento de 10%

Por la compra de una pizza familiar (Q115.00), recibe un descuento de 15%

Por la compra de una pizza fiesta(Q100.00), recibe un descuento de 20%

Solicitar el tipo de pizza, el precio unitario y hallar el precio a pagar.

Grabar los datos a un archivo y mostrar el total global de la venta del dia y

determinar el total de lo vendido por producto, mostrando tambien el porcentaje

con relacion al total.

\*/

#include <fstream>

#include <iostream>

#include <string>

#include <vector>

using namespace std;

//Declaración de estructura de tipo pizza

struct Pizzas{

string tipoPizza;

float precioUnitario;

float descuento;

float precioPagar;

};

//Declaración de Vector de tipo estructura de pizzas

std::vector<Pizzas> vectorPizza;

//Agregar Pizza a Archivo

void agregarPizza(){

/\* Declaración de variables

stPizza es de tipo estructura Pizzas

\*/

Pizzas stPizza;

int tipo;

cout << "--------------------------------------COMPRAR PIZZA--------------------------------------" << endl;

//Seleccionar el tipo de pizza

cout << "Por favor seleccione el tipo de pizza que desea comprar: " << "\n"

<< "1. Pizza Grande ------------------- Q80.00" << "\n"

<< "2. Pizza Familiar ----------------- Q115.00" << "\n"

<< "3. Pizza Fiesta ------------------- Q100.00" << endl;

cout << "Seleccione una opcion (1-3)" << endl;

cin >> tipo;

//Validaciones correspondientes para determinar el descuento y el total a pagar por la pizza

if (tipo == 1){

stPizza.tipoPizza = "Grande";

stPizza.precioUnitario = 80.00;

stPizza.descuento = stPizza.precioUnitario \* 0.1;

stPizza.precioPagar = stPizza.precioUnitario - stPizza.descuento;

}

else if (tipo == 2){

stPizza.tipoPizza = "Familiar";

stPizza.precioUnitario = 115.00;

stPizza.descuento = stPizza.precioUnitario \* 0.15;

stPizza.precioPagar = stPizza.precioUnitario - stPizza.descuento;

}

else if (tipo == 3){

stPizza.tipoPizza = "Fiesta";

stPizza.precioUnitario = 80.00;

stPizza.descuento = stPizza.precioUnitario \* 0.2;

stPizza.precioPagar = stPizza.precioUnitario - stPizza.descuento;

}

//Salida en pantalla de la compra de pizza

cout << "------------------------------------------------------------------" << endl;

cout << "Compra realizada por una piza " << stPizza.tipoPizza << " con un total a pagar de Q" << stPizza.precioPagar << endl;

cout << "------------------------------------------------------------------" << endl;

//Abrir archivo

ofstream archivoPizzas("PIZZAS.txt", std::ios\_base::app);

//Ingresar datos de pizza a archivo

archivoPizzas << stPizza.tipoPizza << "\t" << stPizza.precioUnitario << "\t" << stPizza.precioPagar << "\t" << stPizza.descuento << endl;

//Cerrar archivo

archivoPizzas.close();

return;

}

//Función encargada de extraer datos de archivo

void leerArchivo(){

//Declaración de variable de tipo archivo

ifstream archivoPizzas;

//Declaración de variable de tipo estructura de Pizzas

Pizzas stPizza;

//Abrir archivo

archivoPizzas.open("PIZZAS.txt", ios::in);

//obtener datos desde el archivo

while (archivoPizzas >> stPizza.tipoPizza >> stPizza.precioUnitario >> stPizza.precioPagar >> stPizza.descuento){

//Guardar los datos obtenidos en un vector

vectorPizza.push\_back(stPizza);

}

//Cerrar archivo

archivoPizzas.close();

return;

}

//Función encargada de mostrar el total de la venta del día

void ventaTotalDia(){

/\* Declaración de variables

total: variable que almacenará el total de la venta del día

\*/

float total = 0.00;

//Invocación de la función para extraer los datos del archivo

leerArchivo();

/\*

Iteración de todos las ventas registradas en el vector "vectorPizza"

para realizar la suma correspondiente al total de venta del día

\*/

for (int i = 0; i <= vectorPizza.size(); i ++){

total = total + vectorPizza[i].precioPagar;

}

//Muestra el total de la venta del día

cout << "-------------------------------------------" << endl;

cout <<"El total de la venta del dia es: Q" << total << endl;

cout << "-------------------------------------------" << endl;

}

void totalPorProducto(){

//Declaración de variables

float totalGrande = 0.00;

float descuentoGrande = 0.00;

float totalFamiliar = 0.00;

float descuentoFamiliar = 0.00;

float totalFiesta = 0.00;

float descuentoFiesta = 0.00;

//Invocación de función para extraer datos de archivo

leerArchivo();

//Iteración del vector pizza para determinar el total pagado por pizza y su descuento

for (int i = 0; i < vectorPizza.size(); i ++){

if(vectorPizza[i].tipoPizza == "Grande"){

totalGrande = totalGrande + vectorPizza[i].precioPagar;

descuentoGrande = descuentoGrande + vectorPizza[i].descuento;

}

else if(vectorPizza[i].tipoPizza == "Familiar"){

totalFamiliar = totalFamiliar + vectorPizza[i].precioPagar;

descuentoFamiliar = descuentoFamiliar + vectorPizza[i].descuento;

}

else if(vectorPizza[i].tipoPizza == "Fiesta"){

totalFiesta = totalFiesta + vectorPizza[i].precioPagar;

descuentoFiesta = descuentoFiesta + vectorPizza[i].descuento;

}

}

//Salida

cout << "-------------------------------------------------" << endl;

cout << "PIZZA" << "\t" << "PAGADO" << "\t" << "DESCUENTO" << endl;

cout << "GRANDE" << "\t" << totalGrande << "\t" << descuentoGrande << endl;

cout << "FAM." << "\t" << totalFamiliar << "\t" << descuentoFamiliar << endl;

cout << "FIESTA" << "\t" << totalFiesta << "\t" << descuentoFiesta << endl;

cout << "-------------------------------------------------" << endl;

}

//Función principal del programa

int main(){

//Declaración de variables

char opcion; //Guarda la opción ingresada por el usuario

string vacio; //Variable encargada de obtener un valor vacío para selección de opción en menú

bool terminar = false; //Variable bandera para terminar ejecución de programa

//MENÚ DE PROGRAMA

do {

cin.clear();

cout << "---------------------PIZZERÍA EL GRAN SABOR---------------------" << endl;

cout << "1. Comprar" << "\n"

<< "2. Mostrar total de venta del día" << "\n"

<< "3. Total vendido por producto" << "\n"

<< "4. Salir" << endl;

cout << "Elija una opción (1-4)" << endl;

cin >> opcion;

switch(opcion){

case '1':

agregarPizza();

break;

case '2':

ventaTotalDia();

break;

case '3':

totalPorProducto();

break;

case '4':

terminar = true;

break;

default:

cout << "Opción inválida";

getchar();

getline(cin, vacio);

break;

}

}while(terminar != true);

return 0;

}

/\* EJERCICIO 3

Un empresa que se dedica a la venta de seguros de vehiculos, calcula el importe

según el modelo y color del automovil, asi como la edad del conductor.

Modelo Color Precio

A Blanco 240.50

Metalizado 330.00

Otros 270.50

B Blanco 300.00

Metalizado 360.50

Otros 330.00

Incremento de precio

Edad % Incremento

<31 25%

Entre 31 y 65 0%

>65 30%

Determine el importe a pagar mostrandolo en pantalla según los datos del vehículo (modelo y color)

y la edad del conductor.

Grabe los datos a un archivo.

\*/

#include <fstream>

#include <iostream>

#include <string>

#include <vector>

using namespace std;

//Declaración de estructura de tipo pizza

struct Vehiculos{

int edad;

string modelo;

string color;

float precio;

float importe;

};

//Agregar vehiculo a Archivo

void agregarVehiculo(){

/\* Declaración de variables

stVehiculo es de tipo estructura Vehiculos

\*/

Vehiculos stVehiculo;

int tipo;

int color;

cout << "--------------------------------------REGISTRAR VEHICULO--------------------------------------" << endl;

//Ingresar edad del conductor

cout << "Ingrese la edad del conductor: "<< endl;

cin >> stVehiculo.edad;

//Seleccionar el modelo del vehiculo

cout << "Por favor seleccione el modelo del vehiculo " << "\n"

<< "1. A" << "\n"

<< "2. B" << endl;

cout << "Seleccione una opcion (1-2)" << endl;

cin >> tipo;

//Validaciones correspondientes para determinar el descuento y el total a pagar por la pizza

if (tipo == 1){

stVehiculo.modelo = "A";

cout << "Ingrese color:" << "\n"

<< "1. Blanco-----------Q240.50" << "\n"

<< "2. Metalizado-------Q330.00" << "\n"

<< "3. Otros------------Q270.50" << endl;

cin >> color;

switch (color){

case 1:

stVehiculo.color = "Blanco";

stVehiculo.precio = 240.50;

break;

case 2:

stVehiculo.color = "Metalizado";

stVehiculo.precio = 330.00;

break;

case 3:

stVehiculo.color = "Otros";

stVehiculo.precio = 270.50;

break;

}

}

else if (tipo == 2){

stVehiculo.modelo = "B";

cout << "Ingrese color:" << "\n"

<< "1. Blanco-----------Q300.00" << "\n"

<< "2. Metalizado-------Q360.00" << "\n"

<< "3. Otros------------Q330.00" << endl;

cin >> color;

switch (color){

case 1:

stVehiculo.color = "Blanco";

stVehiculo.precio = 300.00;

break;

case 2:

stVehiculo.color = "Metal.";

stVehiculo.precio = 360.00;

break;

case 3:

stVehiculo.color = "Otros";

stVehiculo.precio = 330.50;

break;

}

}

if (stVehiculo.edad < 31){

stVehiculo.importe = stVehiculo.precio + (stVehiculo.precio \* 0.25);

}

else if(stVehiculo.edad >= 31 and stVehiculo.edad <= 65){

stVehiculo.importe = stVehiculo.precio + (stVehiculo.precio \* 0.00);

}

else if(stVehiculo.edad > 65){

stVehiculo.importe = stVehiculo.precio + (stVehiculo.precio \* 0.30);

}

//Salida en pantalla del registro del vehículo

cout << "------------------------------------------------------------------" << endl;

cout << "El importe a pagar del vehiculo es: Q" << stVehiculo.importe << endl;

cout << "------------------------------------------------------------------" << endl;

//Abrir archivo

ofstream archivoVehiculo("VEHICULOS.txt", std::ios\_base::app);

//Ingresar datos del vehiculo al archivo

archivoVehiculo

<< stVehiculo.edad << "\t" << stVehiculo.modelo << "\t" << stVehiculo.color

<< "\t" << stVehiculo.precio << "\t" << stVehiculo.importe << endl;

//Cerrar archivo

archivoVehiculo.close();

return;

}

int main(){

//Declaración de variables

char opcion; //Guarda la opción ingresada por el usuario

string vacio; //Variable encargada de obtener un valor vacío para selección de opción en menú

bool terminar = false; //Variable bandera para terminar ejecución de programa

//MENÚ DE PROGRAMA

do {

cin.clear();

cout << "---------------------SEGURO VEHICULAR---------------------" << endl;

cout << "1. Importe a pagar" << "\n"

<< "2. Salir" << endl;

cout << "Elija una opción (1-2)" << endl;

cin >> opcion;

switch(opcion){

case '1':

agregarVehiculo();

break;

case '2':

terminar = true;

break;

default:

cout << "Opción inválida";

getchar();

getline(cin, vacio);

break;

}

}while(terminar != true);

return 0;

}