



Pontificia Universidad
JAVERIANA
Bogotá

Facultad de Ingeniería
Departamento de Ingeniería de Sistemas
Introducción a la Programación
Primer Parcial Conjunto – marzo 05 de 2021

Nombre:	<u>Miguel Angel Gonzalez Rodriguez</u>	ID (T.I.):	<u>1109184325</u>
Profesor:	<u>Oscar Danilo Martinez Bernal</u>	Carrera:	<u>Ingenieria Mecatronica</u>

HOJA DE RESPUESTAS

Punto 1.

```
#include <iostream>
using namespace std;
int main(){
    int altura=0, alturaPrev=0, kilometros=0, alturaMax=0, alturaMin=0, montañaAct=-1, montañaAncha=0, valleAct =
-1, valleAncho=0;
    int contValle=0, contMontaña=0, inicio=0; // Ambas lineas solo declare variables, montañaAct y valleAct
inicializan en -1 para que el primer valor ingresado no cuente como kilometro de la montaña o valle
    while(altura >= 0){
        cout << "Altura actual? ";
        cin >> altura; // pregunto la altura actual
        if(inicio == 0 && altura >= 0){ // ese inicio indica si es o no la primera ejecucion para inicializar las 2 variables
            alturaMin = altura;
            alturaMax = altura;
            inicio = 1;
        }
        else if(altura < 0){ // si la altura es negativa se rompe el ciclo
            break;
        }
        else{ // si no se cumple ninguna de las condiciones previas
            if(altura > alturaMax){ // revisa si la altura ingresada es mayor a la maxima guardada
                alturaMax = altura;
            }
            else if(altura < alturaMin){ // revisa si la altura ingresada es menor a la minima guardada
                alturaMin = altura;
            }
            kilometros++; // agrega 1 a la variable que va a indicar los kilometros recorridos
        }
        if(altura > alturaPrev && contMontaña == 0){ // contMontaña es una variable que indica cuando empezó a
bajar o la altura es igual, aqui se revisa si está subiendo una montaña que no ha bajado previamente
            montañaAct++; // montañaAct es una variable que cuenta los kilometros de la montaña que se está
recorriendo
        }
        else if(altura <= alturaPrev){ //revisa si el ciclista está bajando o esta andando en linea recta
            contMontaña=1; // aqui la pongo en 1 porque si vuelve a subir al ser una nueva montaña con esto no va a
ingresar al if previo
        }
    }
}
```



```
montañaAct++;
}
else{ // si empezó una nueva montaña se cumple este else
    contMontaña = 0; // reinicio contador montaña
    if(montañaAct > montañaAncha){ // reviso si el valor de esta montaña fue mayor a la mas ancha guardada
        montañaAncha = montañaAct;
        montañaAct=1; // inicio la distancia de la nueva montaña en 1
    }
    else{ // si esta montaña fue menor a la guardada ejecuto esto
        montañaAct=1;
    }
}
if(altura < alturaPrev && contValle == 0){ // aquí tiene la misma estructura que la montaña pero al ser un valle
revisa si la altura actual es menor a la anterior
    valleAct++; // le sumo 1 a la distancia del valle
}
else if(altura >= alturaPrev){ // si la altura es mayor a la anterior
    contValle = 1; // contValle es = 1 porque si vuelve a bajar es un nuevo valle
    valleAct++;
}
else{ // si empieza nuevo valle
    contValle=0; // reinicio contValle
    if(valleAct > valleAncho){ // si el valle actual fue mas largo que el previamente guardado
        valleAncho = valleAct;
        valleAct=1; // reinicializo valleAct en 1
    }
    else{ // si no fue mayor al guardado
        valleAct=1;
    }
}
alturaPrev = altura; //la altura previa se vuelve la altura actual
}
cout << "Usted recorrio un total de " << kilometros << " kilometros\n";
cout << "Su altura maxima fue de " << alturaMax << " msn\n";
cout << "Su altura minima fue de " << alturaMin << " msn\n";
cout << "La montaña mas ancha tuvo " << montañaAncha << " kilometros\n";
cout << "El valle mas ancho tuvo " << valleAncho << " kilometros\n";
return 0;
}
```



Pontificia Universidad
JAVERIANA
Bogotá

Facultad de Ingeniería
Departamento de Ingeniería de Sistemas
Introducción a la Programación
Primer Parcial Conjunto – marzo 05 de 2021

Punto 2.

```
#include <iostream>
using namespace std;
int main(){
    int numero, caracter1, caracter2, caracter3, caracter4, suma=0, numFinal; // inicializo variables
    string otraVez="si";
    while(otraVez == "si" || otraVez == "Si"){ // hasta que no digan que algo diferente a que si quieren correr el
codigo esto sera infinito
        while(numero > 9999 || numero < 1000){ // reviso si el numero es mayor a 9999 o menor a 1000
            cout << "\nIngrese un numero de 4 cifras: "; // si eso ocurre tiene que ingresar un numero valido
            cin >> numero;
        }
        caracter1 = numero % 10; // saco el ultimo digito y lo almaceno
        numero = numero/10; // remuevo el ultimo digito
        caracter2 = (numero % 10); // se aquí hasta el caracter 4 es el mismo proceso
        numero = numero/10;
        caracter3 = (numero % 10);
        numero = numero / 10;
        caracter4 = (numero % 10);
        suma = caracter1 + caracter2 + caracter3 + caracter4; // sumo los digitos
        numFinal= caracter1 + caracter2 * 10 + caracter3 * 100 + caracter4 * 1000; // vuelvo a crear el numero original
pero en la variable numFinal
        if(suma % 2 == 0){ // reviso si la suma de los digitos es par
            numFinal = numFinal * 10 + 1; // si es par a numFinal le agrego el 1 al final multiplicandolo por 10 y
sumandole 1
        }
        else{ // reviso si es impar
```



Pontificia Universidad
JAVERIANA
Bogotá

Facultad de Ingeniería
Departamento de Ingeniería de Sistemas

Introducción a la Programación

Primer Parcial Conjunto – marzo 05 de 2021

```
        numFinal = numFinal * 10; // lo mismo del par pero sin sumar 1 por lo tanto se le agrega un 0
    }
    cout << "El numero de la tarjeta seria: " << numFinal;
    cout << "\nDeseas probar otro numero?: (si/no) ";
    cin >> otraVez; // reviso si quiere hacer esto otra vez o no
}
return 0;
}
```