



UNLaM

Dto. Ingeniería e Investigaciones Tecnológicas

(1110) PROGRAMACIÓN

Parcial 2

Turno TARDE

Ma– Vi 14-18 TT

22/11/2022

Apellido y Nombre:

DNI:

PARCIAL:

Calificación:

TEMA A

Ejercicio 1:

Se dispone de un archivo binario “mediciones.dat” que contiene los valores de temperatura capturados de distintos dispositivos. Cada registro tiene un entero conteniendo el número de dispositivo y otro entero donde se almacena la temperatura en valor crudo. Para obtener la temperatura en grados celsius (valor de ingeniería) es necesario multiplicar el crudo por una ganancia y sumar un desplazamiento (calibración). Los coeficientes de calibración para cada dispositivo están en el archivo “coeficientes.txt”.

Se pide, leyendo una única vez cada archivo, presentar por pantalla el promedio de mediciones para cada dispositivo en valores de ingeniería (grados celsius). Utilice la o las estructuras de datos que le resulte más conveniente para implementar lo solicitado.

Para promocionar el examen implemente las primitivas de forma recursiva (Sin bucles while/for).

Nota: No se conoce de antemano la cantidad de dispositivos, no la suponga y recuerde que no se permite VLA (Variable Length Array), $a[x]$ donde x es determinado en tiempo de ejecución.

Ejercicio 2 C++:

Desarrolle FVector (Vector de flotantes) para que el código “main” provisto con el examen sea válido. Al pie la salida esperada. Tenga en cuenta que los objetos de la clase FVector almacenan flotantes de manera contigua y de cantidad variable. La cantidad máxima de flotantes que se pueden almacenar viene dada por las capacidades de memoria del equipo y no por una limitación de la clase FVector teniendo en cuenta que tampoco se permite VLA. Sea cuidadoso con el manejo de memoria, analice la salida por pantalla para inferir el significado de las operaciones realizadas y no implemente métodos o constructores públicos innecesarios.

```
int main()
{
    float av1[] = {0.0, 1.0, 2.0, 3.0};
    float av2[] = {4.0, 5.0, 6.0, 7.0};
    float av3[] = {1.0, 2.0};

    FVector fv1(av1, sizeof(av1)/sizeof(float));
    FVector fv2(av2, sizeof(av2)/sizeof(float));
    FVector fv3(av3, sizeof(av3)/sizeof(float));

    if(fv2==(fv1+=4.0)){
        cout<<"Los vectores fv1 y fv2 son iguales: "<<endl;
        cout<<fv1<<" = "<<fv2<<endl;
    }
    cout<<fv2--<<endl;
    if(fv2==fv1){
        //No aplica a la prueba
    }else{
        cout<<"Los vectores fv1 y fv2 ahora son distintos: "<<endl;
        cout<<fv1<<" != "<<fv2<<endl;
    }
    try{
        cout<<fv1/fv3<<endl;
    }catch(exception& ex){
        cout<<ex.what()<<endl;
    }

    cout<<"Division de vectores fv1 y fv2: "<<fv1/fv2<<endl;
    return 0;
}
```

```
Los vectores fv1 y fv2 son iguales:
{4,5,6,7} = {4,5,6,7}
{4,5,6,7}
Los vectores fv1 y fv2 ahora son distintos:
{4,5,6,7} != {3,4,5,6}
Excepcion: Este operador no aplica a vectores de diferente longitud...
Division de vectores fv1 y fv2: {1.33333,1.25,1.2,1.16667}
```

```
Process returned 0 (0x0)   execution time : 0.078 s
Press any key to continue.
```

EVALUACIÓN

NOTA GENERAL

- La hora límite de entrega 17:45 hs.
- Las notas serán enviadas por MIEL
- Desarrolle cada ejercicio en un proyecto separado.
- Para el ejercicio de C++ y para simplificar la corrección desarrolle todo en el mismo proyecto, no haga una biblioteca aparte
- **Incluya en el encabezado de cada archivo, // apellido_nombre_DNI**
- Recuerde antes de comprimir, eliminar las carpetas bin y obj de cada proyecto.
- **Entregue ambos proyectos compactados en un único archivo zip. El nombre del archivo debe tener el siguiente formato “apellido_nombre_DNI.zip”.**
- Entregue el parcial usando prácticas de MIEL.
- Enviar a todos los tutores.
- ¡La evaluación es individual!

¡El mayor de los éxitos!