

Ejercitación Clase Práctica 1

Programación en C++

Ejercicio 1: Compilación y ejecución de un programa en C++

A lo largo de esta ejercitación, junto con la correspondiente de la siguiente clase, se programará el *Sistema SUVE* (Sistema único de viaje electrónico).

En este sistema tendremos personas que viajan y poseen una tarjeta electrónica. Y dispositivos electrónicos que leen y procesan estas tarjetas.

01_iniciando_suve

Lo primero que hará una terminal de SUVE, es mostrar un mensaje de inicio cuando una persona va a usar el sistema.

- Usar la opción “Open Folder” y abrir la carpeta “Ejercicio1”.
- Abrir el archivo “01_iniciando_suve.cpp”.
- Compilar el archivo usando la orden “Tasks: Run build task”.
 - Para ejecutar una orden en VSCode, usar el shortcut “Ctrl+Shift+p”. Elegir “clang++” o “g++” como compilador.
 - Alternativamente, puede ejecutar la orden requerida con el shortcut “Ctrl+Shift+b”.
- Abrir una terminal en VSCode y ejecutar el archivo recién creado “01_iniciando_suve”.
 - Para ello se puede usar la orden: `./01_iniciando_suve`
 - *Aclaración:* “./” busca en el directorio donde está ubicada la línea de comandos, por lo que buscará el archivo 01_iniciando_suve en ese directorio. Por defecto, la terminal está en el directorio del proyecto (si se abrió una terminal bash desde VSCode).
- Modificar el código imprimiendo el mensaje de inicio 5 veces. Extraer la instrucción completa a una función para evitar repetir código.
 - El formato de una función c++ es:

```
tipo_de_dato nombre_de_la_funcion(tipo_de_dato parametro1, ...) {  
    ...código...  
}
```

- La función a extraer no devuelve ningún valor. El tipo `_de_dato` para representar esta idea es “void”.
- ¿qué sucede si escribimos la función después de su uso?
- Separar la definición de la implementación. Ubicar la implementación a continuación del main.

02_datos_random

- Abrir el archivo “02_datos_random.cpp” y ejecutarlo (Ctrl+Shift+b -> ejecutar en terminal: `./02_datos_random`).
- Extraer las variables a funciones que devuelvan los respectivos valores.
- ¿qué ocurre si la variable y la función tienen el mismo nombre?

03_sucesiones

- Abrir el archivo “03_sucesiones.cpp” y ejecutarlo.
- Completar el código para que el usuario pueda ingresar un numero.
 - Usar la función “`std::cin` » variable”.
- Implementar una función que dado un i , devuelva el i ésimo término de la sucesión de Fibonacci.

Se dice que en el siglo 12, el matemático italiano Leonardo de Piso (alias fibonacci) planteó la sucesión que lleva su mismo nombre.

Esta sucesión dice que para obtener su i ésimo término, que lo llamaremos F_i , debemos sumar los 2 términos anteriores, empezando con 0 y 1. Es decir:

$$F_0 = 0$$

$$F_1 = 1$$

$$F_i = F_{i-1} + F_{i-2}$$

Un pseudocódigo para obtener el i ésimo término es:

```
anterior = 0
actual = 1
si i = 0
    devolver 0
si i = 1
    devolver 1
Para x = 2 hasta i
    nuevo_actual = actual + anterior
    anterior = actual
    actual = nuevo_actual
devolver actual
```

Tip: Si se escribe la sentencia “using namespace std;” al principio del archivo, se puede evitar escribir “std::” en el archivo.

- Crear un nuevo archivo “sucesion_de_fibonacci.h”.
- Mover la definición de la función “sucesion_de_fibonacci” al nuevo archivo.
- Incluir el contenido del archivo en “03_sucesiones.cpp” usando la directiva:
`#include "nombre_del_archivo"`
- Crear un nuevo archivo “sucesion_de_fibonacci.cpp”.
- Mover la implementación de la función al nuevo archivo.
- ¿Qué ocurre si se intenta compilar al igual que siempre?
- Abrir una terminal y compilar con:

```
g++ -o nombre_archivo_ejecutable 03_sucesiones.cpp sucesion_de_fibonacci.cpp
```

Ejercicio 2: separar código en varios archivos

Uso de Cmake

- Abrir el proyecto Ejercicio2. Recordar de abrir “New window” para crear una nueva conexión de docker con su respectivo compilador y herramientas. Luego abrir con “Open folder”.
- Configurar Cmake.
- Usar botón “build” de la barra inferior para compilar el proyecto.
- Ejecutar el programa por medio del símbolo de play de la barra inferior.