UNIVERSIDAD DE COSTA RICA
ESCUELA DE CIENCIAS DE LA
COMPUTACIÓN E INFORMÁTICA

CI-0118 Lenguaje Ensamblador
Dr. Carlos Vargas
I Ciclo del 2024

Segunda Tarea Programada

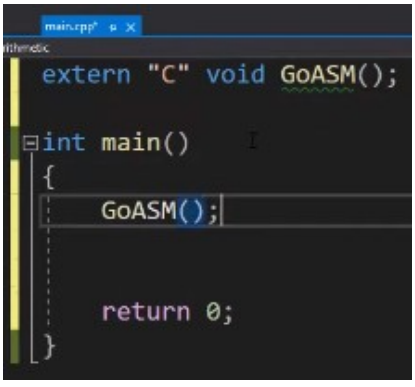
Programar la simulación de una "Calculadora Vectorial" que opere números reales, utilizando programación híbrida (mezcla de lenguaje Visual C++ y ensamblador), corriendo en modo protegido**, moderna arquitectura Intel X64.**
Para desarrollar la aplicación se utiliza el Ambiente de Desarrollo Integrado(IDE) **"Visual Studio Community"** versión actual, el cual ya incluye al MASM.

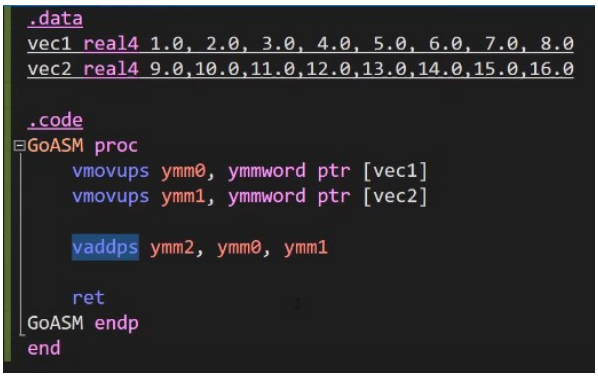
Esta tarea se desarrolla en equipos, conformados por el mismo grupo de estudiantes del proyecto de investigación. Debe incluir código fuente (el project) y documentación externa en formato .PDF. Fecha de entrega: definida en carta al estudiante.

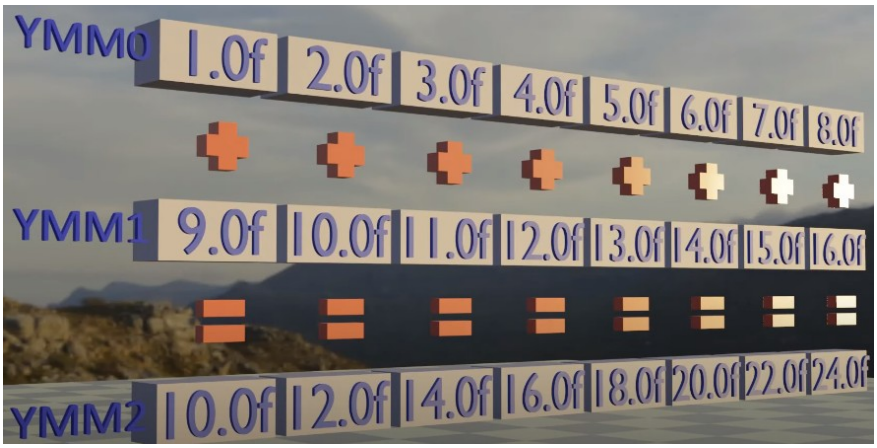
La calculadora efectúa dos operaciones:

1. Suma de dos vectores.
2. Multiplicación de un escalar por los respectivos valores de un vector.

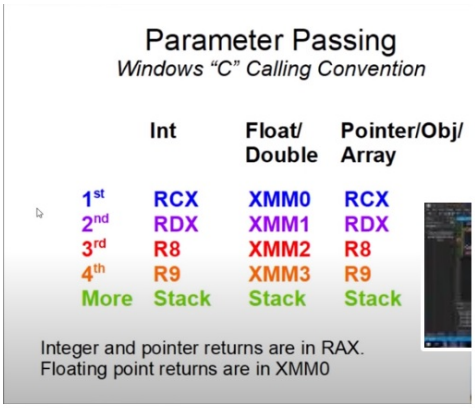
La interfaz de usuario se codifica en C++ y las operaciones mediante procs en ensamblador, explotando al máximo las capacidades nativas de la arquitectura AVX de procesamiento en paralelo, usando registros de 128 bytes (XMM), o de 256 bytes (YMM) o de 512 bytes (ZMM). A continuación un ejemplo de procedimiento en ensamblador que suma dos vectores.







Tomar en consideración las convenciones para pasar parámetros:

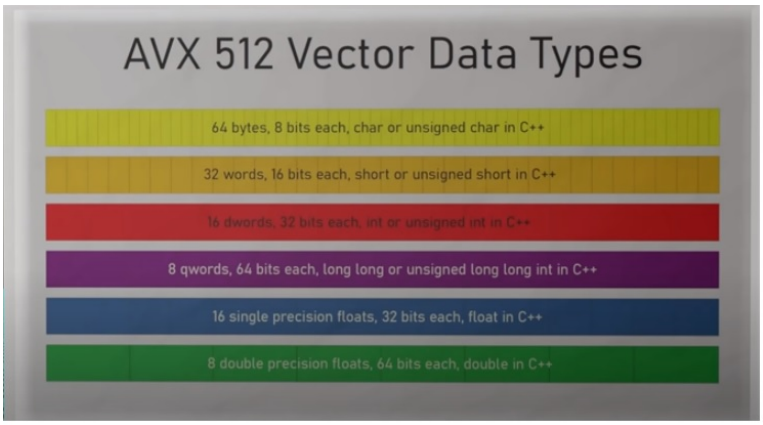


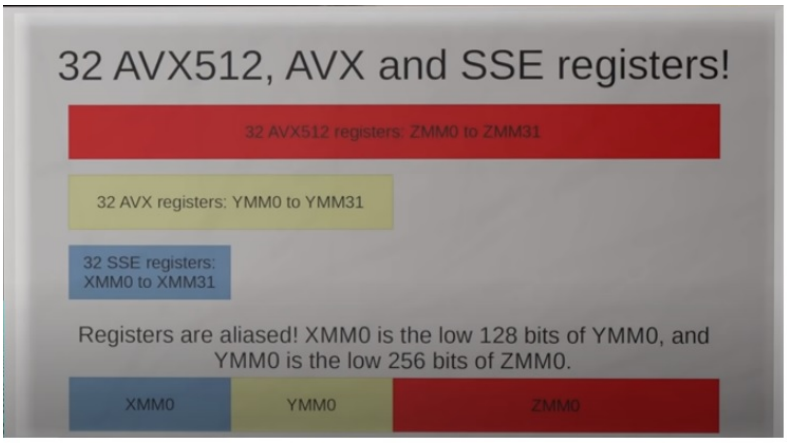
Para garantizar que la CPU soporta la arquitectura AVX 2, se recomienda hacer la comprobación, con el siguiente código:

```
#include <iostream>

extern "C" void AVX512Test(double* c, double* a, double* b);

extern "C" bool AVXFoundationDetection();

int main()
{
    if (AVXFoundationDetection())
        std::cout << "This CPU is capabale of ZVX512 Fondation instruction set!" << std::endl;
    else
        std::cout << "Nope" << std::endl;
```

```
.code
AVXFoundationDetection proc
    push rbx

    mov eax, 7
    mov ecx, 0

    cpuid

    shr ebx, 16
    and ebx, 1

    mov eax, ebx

    pop rbx
    ret
AVXFoundationDetection endp
```

Recuerde que los registros se traslapan (comparten una misma área):





Para la evaluación de esta tarea, se aplicarán los siguientes porcentajes: documentación interna 10%, documentación externa 10% y 80% el resto. Los siguientes puntos sirven como **una guía** para la documentación externa:

1. Descripción del problema.
2. Algoritmos utilizados.
3. Estructura del programa:
    1. Descripción de constantes y de variables.
    2. Procedimientos.
4. Detalles importantes de la implementación.
5. Datos de prueba utilizados.
6. Análisis de resultados (indicando grado de funcionamiento)
7. Breve Guía del Usuario.