Name: Marlon Morales

Activity 1 - Interpretation of Directives

Directive	Description
.syntax unified	Specifies the use of the unified assembly syntax, which allows for a single syntax style compatible with both ARM and Thumb instructions
.cpu cortex-m4	Selects the Cortex-M4 as the target processor. This informs the assembler to support instructions and architectural features specific to the ARM Cortex-M4 CPU. It also clears any previously set architecture extensions, ensuring the assembly code strictly conforms to the selected CPU.
.fpu softvfp	Indicates that software floating-point operations (not hardware FPU) should be used. This is useful when the target hardware lacks a hardware FPU or when using a toolchain that emulates floating-point in software.
.thumb	Instructs the assembler to generate Thumb (16-bit compressed) instruction set code instead of full 32-bit ARM instructions. Cortex-M processors execute only Thumb code.
.section .data	Begins a new section in the binary for data variables. The .data section typically contains initialized global and static variables.
.balign	Aligns the following data on a specified byte boundary (often 4 or 8 bytes). Ensures proper memory alignment for performance and correctness.
array: .word 1,2,3,4,5,6,7,8,9,10,-1	Defines a label array and allocates 11 consecutive 4-byte words in memory initialized with the given integers. The (-1) signifies the end of the array.
.section .text	Starts the code section of the program (often called the *text segment*), where executable instructions are stored. This is where the main logic of the program resides.
.balign	Aligns the following data on a specified byte boundary (often 4 or 8 bytes). Ensures proper memory alignment for performance and correctness.
.global main	Declares the main label as global, making it visible to the linker and other files. This allows the linker to recognize main as the entry point of the program.