# **CS 410 Binary to Assembly Activity Template**

**Step 1:** List the binary file name.

**Step 2:** Identify the functions in the binary file.

**Step 3**: Convert the binary file to assembly code.

**Step 4:** Align the blocks of assembly code with their corresponding function in the binary file.

**Step 5:** Explain the functionality of the blocks of assembly code.

**File One: assignment3\_1.o**

|  |  |  |
| --- | --- | --- |
| **Functions** | **Blocks of Assembly Code** | **Explanation of Functionality** |
| \_init | 400418: sub $0x8, %rsp 400431: retq | Initializes the program by calling \_\_gmon\_start\_\_ if available and setting up the stack. |
| \_start | 400490: xor %ebp, %ebp 4004b4: callq \_\_libc\_start\_main@plt | Program entry point, initializes registers and calls \_\_libc\_start\_main, which leads to main. |
| deregister\_tm\_clones | 4004c0: mov $0x60104f, %eax 4004e2: jmpq \*%rax | Manages transactional memory clone deregistration based on clone existence. |
| register\_tm\_clones | 4004f0: mov $0x601048, %eax 400527: jmpq \*%rdx | Registers transactional memory clones, setting up necessary memory operations. |
| \_\_do\_global\_dtors\_aux | 400530: cmpb $0x0, 0x200b11(%rip) 40054a: retq | Handles global destructors by calling deregister\_tm\_clones and setting a completion flag. |
| frame\_dummy | 400550: cmpq $0x0, 0x2008c8(%rip) 400578: jmp register\_tm\_clones | Initializes register\_tm\_clones if needed. |
| main | 40057d: push %rbp 4005a4: callq exit@plt | The main function displays text using puts, and calls exit to terminate the program. |
| \_\_libc\_csu\_init | 4005b0: push %r15 400614: retq | Called during initialization to set up parameters and call routines for \_\_libc\_start\_main. |
| \_\_libc\_csu\_fini | 400620: retq | Final cleanup before program termination. |
| \_fini | 400624: sub $0x8, %rsp 40062c: retq | Cleans up the stack and finalizes program exit. |

## File Two: assignment3\_2.o

|  |  |  |
| --- | --- | --- |
| Functions | Blocks of Assembly Code | Explanation of Functionality |
| \_init | 4004a8: sub $0x8, %rsp 4004c1: retq | Initializes the program by calling \_\_gmon\_start\_\_ if available. |
| \_start | 400540: xor %ebp, %ebp 400564: callq \_\_libc\_start\_main@plt | Sets up the initial state and calls \_\_libc\_start\_main, which ultimately calls main. |
| deregister\_tm\_clones | 400570: mov $0x60105f, %eax 400599: jmp \*%rax | Manages deregistration of transactional memory clones, conditionally freeing clones if they exist. |
| register\_tm\_clones | 4005a0: mov $0x601058, %eax 4005d7: jmp \*%rdx | Registers transactional memory clones by adjusting memory for the program. |
| \_\_do\_global\_dtors\_aux | 4005e0: cmpb $0x0, 0x200a71(%rip) 4005fa: retq | Handles global destructors by calling deregister\_tm\_clones and marking cleanup completion. |
| frame\_dummy | 400600: cmpq $0x0, 0x200818(%rip) 400628: jmp register\_tm\_clones | Sets up register\_tm\_clones initialization if required. |
| main | 40062d: push %rbp 40067f: callq exit@plt | The main function prompts for input, calls isoc99\_scanf for user input, and displays results using printf. |
| \_\_libc\_csu\_init | 400690: push %r15 4006f4: retq | Initialization routine, ensuring all parameters are set for \_\_libc\_start\_main. |
| \_\_libc\_csu\_fini | 400700: retq | Final cleanup before program termination. |
| \_fini | 400704: sub $0x8, %rsp 40070c: retq | Finalizes the program by cleaning up the stack. |

**File Three: assignment3\_3.o**

|  |  |  |
| --- | --- | --- |
| **Functions** | **Blocks of Assembly Code** | **Explanation of Functionality** |
| \_init | 4004a8: sub $0x8, %rsp 4004c1: retq | Initializes the program, calling \_\_gmon\_start\_\_ if available. |
| \_start | 400540: xor %ebp, %ebp 400564: callq \_\_libc\_start\_main@plt | Entry point, setting up and calling \_\_libc\_start\_main, which invokes main. |
| deregister\_tm\_clones | 400570: mov $0x60105f, %eax 400599: jmp \*%rax | Frees transactional memory clones if they exist, conditional on clone availability. |
| register\_tm\_clones | 4005a0: mov $0x601058, %eax 4005d7: jmp \*%rdx | Adjusts pointers for clone registration, setting up required memory allocations. |
| \_\_do\_global\_dtors\_aux | 4005e0: cmpb $0x0, 0x200a71(%rip) 4005fa: retq | Calls deregister\_tm\_clones and marks completion. |
| frame\_dummy | 400600: cmpq $0x0, 0x200818(%rip) 400628: jmp register\_tm\_clones | Checks if clone registration setup is necessary. |
| AddNumbers | 40062d: push %rbp 400640: retq | Adds two numbers passed as arguments and returns the result in %eax. |
| main | 400641: push %rbp 40069e: callq exit@plt | Main function calls AddNumbers, displays the result, and exits. |
| \_\_libc\_csu\_init | 4006b0: push %r15 400714: retq | Sets up parameters for \_\_libc\_start\_main during initialization. |
| \_\_libc\_csu\_fini | 400720: retq | Cleans up before program termination. |
| \_fini | 400724: sub $0x8, %rsp 40072c: retq | Final stack cleanup. |

**File Four: assignment3\_4.o**

|  |  |  |
| --- | --- | --- |
| **Functions** | **Blocks of Assembly Code** | **Explanation of Functionality** |
| \_init | 4004a8: sub $0x8, %rsp 4004c1: retq | Initializes by calling \_\_gmon\_start\_\_ if available. |
| \_start | 400540: xor %ebp, %ebp 400564: callq \_\_libc\_start\_main@plt | Sets up entry point and calls \_\_libc\_start\_main. |
| deregister\_tm\_clones | 400570: mov $0x60105f, %eax 400599: jmp \*%rax | Frees transactional memory clones if they are present. |
| register\_tm\_clones | 4005a0: mov $0x601058, %eax 4005d7: jmp \*%rdx | Registers transactional memory clones, adjusting memory allocations. |
| \_\_do\_global\_dtors\_aux | 4005e0: cmpb $0x0, 0x200a71(%rip) 4005fa: retq | Calls deregister\_tm\_clones and marks as complete. |
| frame\_dummy | 400600: cmpq $0x0, 0x200818(%rip) 400628: jmp register\_tm\_clones | Initializes register\_tm\_clones if required. |
| PrintFact | 40062d: push %rbp 400687: retq | Computes the factorial of a number by multiplying sequentially and displaying intermediate steps. |
| PrintSum | 400688: push %rbp 4006de: retq | Computes the sum of numbers up to a given integer and displays intermediate steps. |
| DisplayMenu | 4006df: push %rbp 400716: retq | Prints a menu for user interaction, showing options available. |
| main | 400717: push %rbp 4007ad: callq exit@plt | Displays the menu, handles user input, and calls PrintFact or PrintSum based on user choice. |
| \_\_libc\_csu\_init | 4007c0: push %r15 400824: retq | Sets up initial parameters and calls routines for \_\_libc\_start\_main. |
| \_\_libc\_csu\_fini | 400830: retq | Cleans up before program termination. |
| \_fini | 400834: sub $0x8, %rsp 40083c: retq | Cleans up the stack and finalizes exit. |