# CS 410 Project One Proficiency Test Template

## Explain the functionality of the blocks of assembly code.

**Main Function**

|  |  |
| --- | --- |
| **Assembly Code Block** | **Explanation of Functionality** |
| 0: push %rbp | Saves the base pointer of the previous stack frame. |
| 1: mov %rsp,%rbp | Sets up the new base pointer for the current stack frame. |
| 4: lea 0x0(%rip),%rsi | Loads the effective address to %rsi (used for parameters). |
| 12: callq 17 <main+0x17> | Calls another function (possibly a subroutine, could be CheckUserPermissionAccess). |
| 1c: mov %eax,0x0(%rip) | Moves the result into memory at the computed address, possibly for further use in the program. |
| 28: cmp $0x1,%eax | Compares the result stored in %eax with 1. |
| 2b: je 40 <main+0x40> | Jumps to address 0x40 if the comparison with 1 is true. |
| 4b: jmp 17 <main+0x17> | Jumps back to re-call or loop through the earlier section. |
| 81: callq 86 <main+0x86> | Calls another function based on a previously calculated address. |
| b3: callq ba <main+0xba> | Further function calls for subroutines or logical flows based on conditions. |
| f2: callq f7 <main+0xf7> | A possible jump to handle termination or reset if conditions match. |
| 11f: retq | Ends the function and returns control to the caller. |

**ChangeCustomerChoice Function**

|  |  |
| --- | --- |
| **Assembly Code Block** | **Explanation of Functionality** |
| 42d: push %rbp | Sets up the function stack frame by pushing the base pointer. |
| 42e: mov %rsp,%rbp | Moves the stack pointer to the base pointer. |
| 478: callq 47d <ChangeCustomerChoice+0x50> | Calls a function within this context, likely changing or setting the customer’s choice. |
| 4f8: retq | Completes and returns from the function to the previous caller. |

**CheckUserPermissionAccess Function**

|  |  |
| --- | --- |
| **Assembly Code Block** | **Explanation of Functionality** |
| 120: push %rbp | Sets up the stack frame for CheckUserPermissionAccess. |
| 124: push %rbx | Further stack frame setup by saving %rbx. |
| 129: sub $0x48,%rsp | Allocates space on the stack. |
| 13f: callq 144 <CheckUserPermissionAccess+0x24> | Calls an internal subroutine, likely checking permissions. |
| 16e: lea 0x0(%rip),%rsi | Loads an address into %rsi, used for permission verification. |
| 1f0: retq | Ends CheckUserPermissionAccess function, returning control. |

**DisplayInfo Function**

|  |  |
| --- | --- |
| **Assembly Code Block** | **Explanation of Functionality** |
| 241: push %rbp | Sets up the stack frame for DisplayInfo. |
| 24c: lea 0x0(%rip),%rsi | Loads the address into %rsi, possibly indicating information to display. |
| 30a: callq 30f <DisplayInfo+0x60> | Calls subroutine to fetch or print the information to be displayed. |
| 42b: retq | Returns from DisplayInfo, completing this function. |