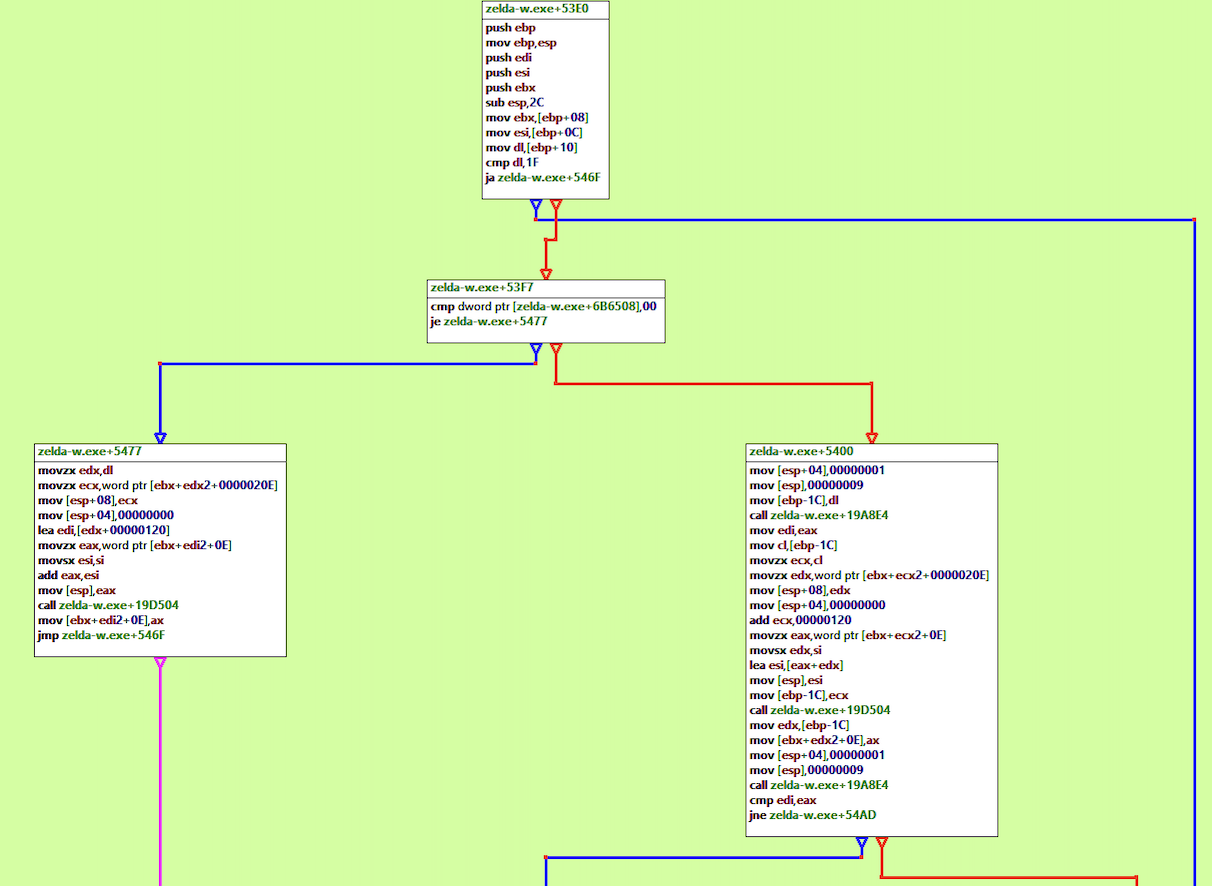
**Task 3**

**Set Rupee to 255 when any rupee is collected**

**Team Members: Sayed Md Abu, Siyu Deng, Ige Tosin O**

To accomplish this task, we need to find which instructions are responsible for rupee collection. Based on previous discovery, we already know the instruction which updates the value of rupee is locating at 405452 which belongs to function starts from 53E0.

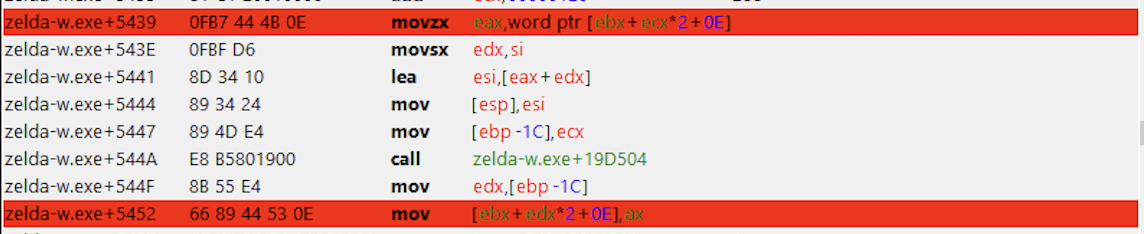


This instruction update our rupee value by copy value from ax register. And we also noticed that if we collect a rupee with value 5, this instruction will run 5 times to update the total rupee value. Then we will concern what will happen if we directly assign 255 to the target address right here.

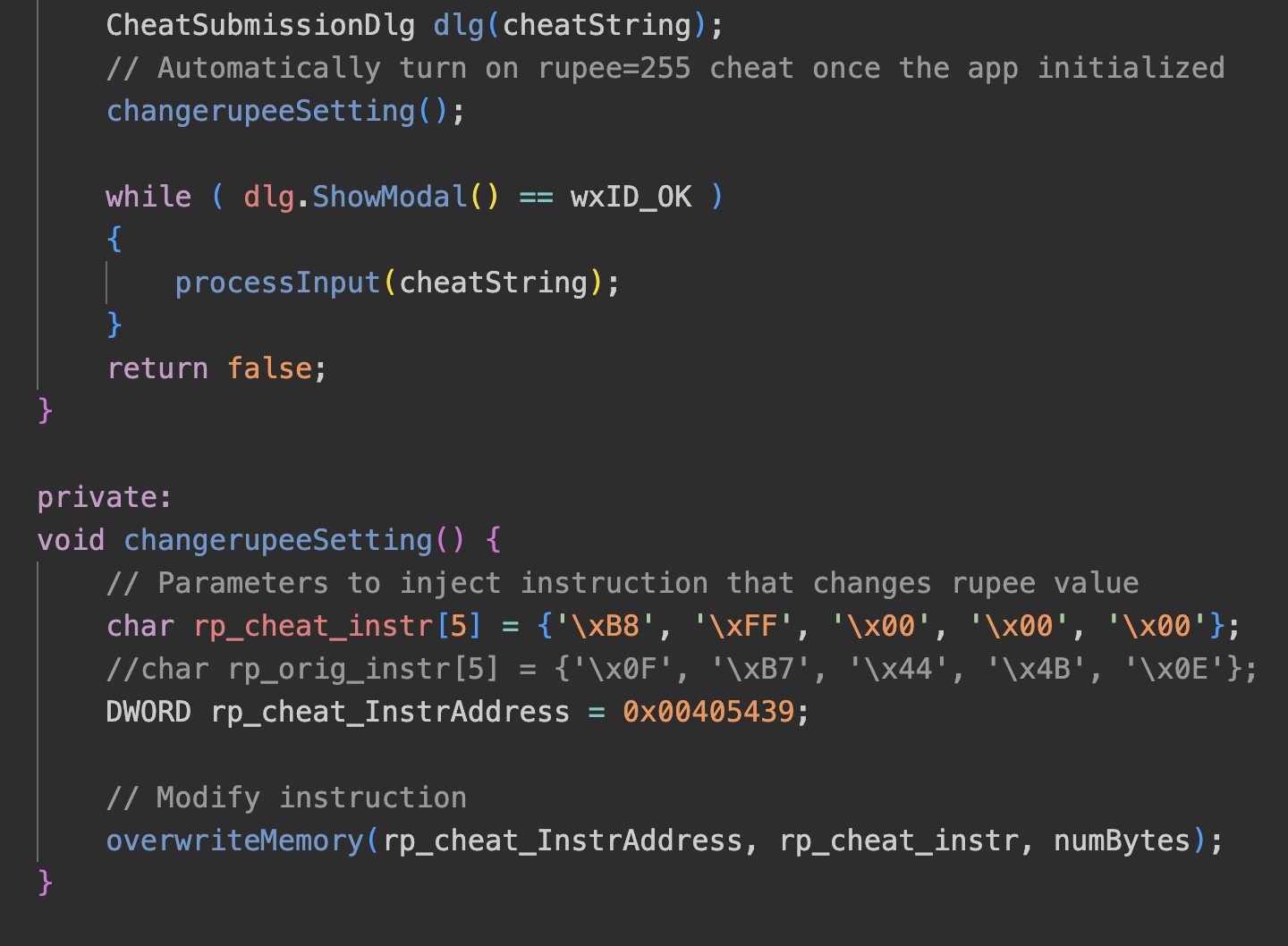
Will collect value 5 rupee make our total rupee value increased by 5\*255 instead of 255?

The answer is no, since ax is a one byte register. The maximum value our rupee could be is 255.

After this reasoning, we know our task can be accomplished by modifying the value carried by ax. However, we soon noticed direct modification on this instruction is not doable. Since this instruction “mov [ebx+edx\*2+0E], ax” is shorter than the instruction we would like to insert “mov [ebx+edx\*2+0E], ff”. Forcefully inject will cause stack overflow, and the overruns hex codes will overwrite the following instruction. This will make the whole program crashed.

After we explored near instructions, we find there is another instruction at address 00405439 we can modify which will eventually change ai’s value at the target instruction.

We modify this instrution “movzx eax, word ptr [ebx+ecx\*2+0E]” to “mov eax,ff”. And our code looks like the following:



And this functionality will be automatically enabled once the dll is injected.

Injectee.cpp is stored at **./Injectee/Injectee.cpp**.