Checkpoint B

In this phase of the project, you will extend code written in checkpoint A by adding more functions and performing more operations on weapon inventory.

Write a function addWeapon to add a specific weapon to the current inventory.

Your function must take the current weapon inventory, and the Weapon type to be added as parameters.

Write a function removeWeapon to remove a specific weapon from the current inventory.

Your function must take the current weapon inventory, and the Weapon type to be removed as parameters.

Write a function toggleWeapon to toggle a specific weapon from the current inventory.

Your function must take the current weapon inventory, and the Weapon type to be toggled as parameters.

In the main function, continue onwards from checkpoint A where you displayed the current weapon inventory and perform the following operations:

Toggle the weapon Poison from the current inventory and display the inventory.

Toggle the weapon Fire from the current inventory and display the inventory.

Check if the weapon Ice exists in the current inventory. If so, remove it from the inventory otherwise print a message indicating that the weapon doesn't exist (see sample output below). Then, display the inventory.

Check if the weapon Melee does not exist in the current inventory. If so (i.e. if it doesn't exist), add it to the inventory otherwise print a message indicating that the weapon already exists (see sample output below). Then, display the inventory.

Print out the final value of the variable representing the inventory (myweapon) as an integer and in binary (see sample output below).

Sample Output for Final code (note: value for myweapon is generated randomly)

Starting with myweapon = 12

Binary representation: 00001100

Your Weapons: Ice Poison

\*\*\*Toggling Poison\*\*\*

Your Weapons: Ice

\*\*\*Toggling Fire\*\*\*

Your Weapons: Fire Ice

\*\*\*Found Ice.. removing it\*\*\*

Your Weapons: Fire

\*\*\*Melee not found.. adding it\*\*\*

Your Weapons: Melee Fire

Ending with myweapon = 3

Binary representation: 00000011

Starting with myweapon = 9

Binary representation: 00001001

Your Weapons: Melee Poison

\*\*\*Toggling Poison\*\*\*

Your Weapons: Melee

\*\*\*Toggling Fire\*\*\*

Your Weapons: Melee Fire

\*\*\*Ice not found.. can't remove it\*\*\*

\*\*\*Melee already exists.. can't add it\*\*\*

Ending with myweapon = 3

Binary representation: 00000011

Starting with myweapon = 4

Binary representation: 00000100

Your Weapons: Ice

\*\*\*Toggling Poison\*\*\*

Your Weapons: Ice Poison

\*\*\*Toggling Fire\*\*\*

Your Weapons: Fire Ice Poison

\*\*\*Found Ice.. removing it\*\*\*

Your Weapons: Fire Poison

\*\*\*Melee not found.. adding it\*\*\*

Your Weapons: Melee Fire Poison

Ending with myweapon = 11

Binary representation: 00001011

Starting with myweapon = 10

Binary representation: 00001010

Your Weapons: Fire Poison

\*\*\*Toggling Poison\*\*\*

Your Weapons: Fire

\*\*\*Toggling Fire\*\*\*

Your Weapons: None

\*\*\*Ice not found.. can't remove it\*\*\*

\*\*\*Melee not found.. adding it\*\*\*

Your Weapons: Melee

Ending with myweapon = 1

Final code

In this phase of the project, you will add a couple of functions to your checkpoint B code and submit your code for grading.

Write a function getWeaponTypeFromString with the following prototype

Weapon getWeaponTypeFromString(string weaponName)

It takes a string as an input parameter and returns the corresponding Weapon type. For example, if weaponName is "Melee", it returns Weapon::Melee. The function works irrespective of the case of the letters in weaponName.

If the user provides a weapon that is not among the 4 weapons that our assassin's inventory can hold, then the function returns Weapon::none

Write a function loadWeapons that prompts the user to enter several weapons in a single line.

It does not take any parameters.

It reads each weapon name from that line, gets its weapon type, and adds that weapon to the inventory.

It returns the unit8\_t variable representing weapon inventory.

Calls the function getWeaponTypeFromString to get the weapon type from weapon name.

In the main function replace the random assignment to the variable myweapon (in checkpoint B) with a call to the function loadWeapons. (Essentially, this replaces random weapon inventory to the weapon inventory indicated by the user).