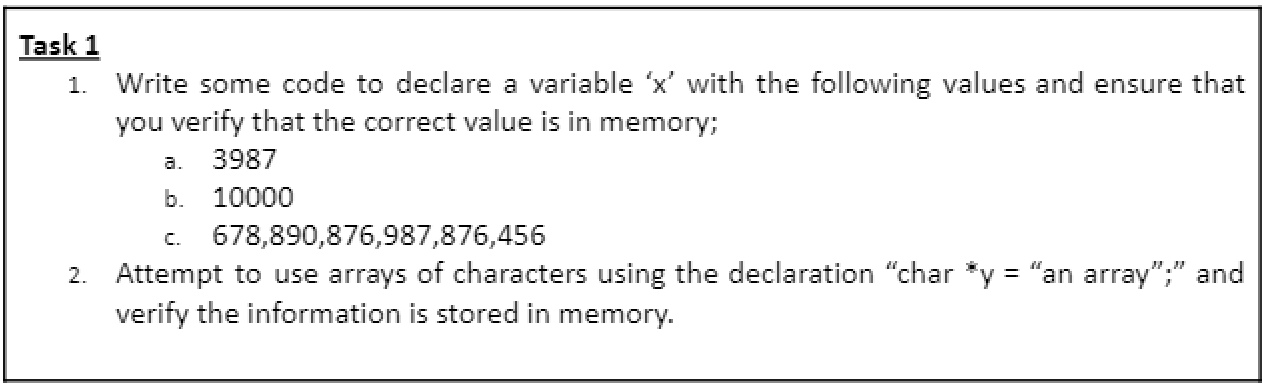
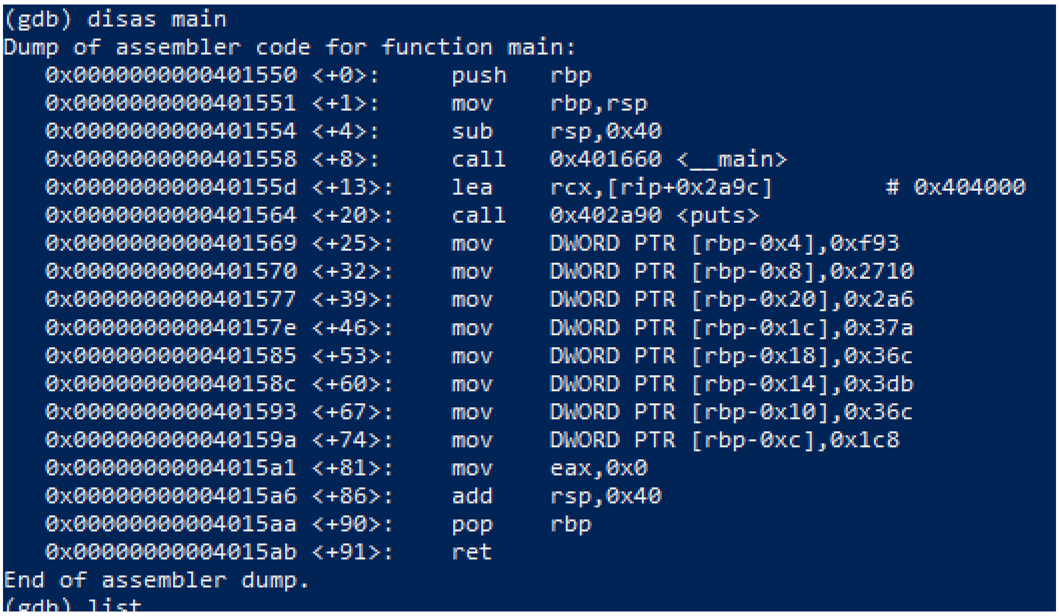
Practicle - gdb



Task 1

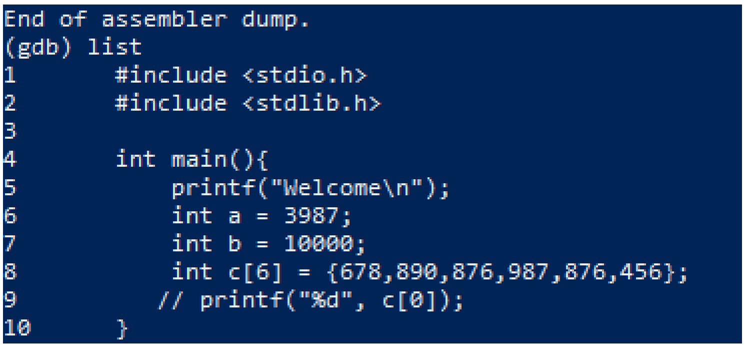
1. Write some code to declare a variable 'x' with the following values and ensure that you verify that the correct value is in memory;
   a. 3987
   b. 10000
   c. 678,890,876,987,876,456
2. Attempt to use arrays of characters using the declaration "char *y = "an array";" and verify the information is stored in memory.

as seen bellow, the memory dumb shows the values being assigned to memory. if you convert the hex values being moved into memory you will see they are the same.



```
(gdb) disas main
Dump of assembler code for function main:
   0x0000000000401550 <+0>:     push   rbp
   0x0000000000401551 <+1>:     mov    rbp,rsp
   0x0000000000401554 <+4>:     sub    rsp,0x40
   0x0000000000401558 <+8>:     call   0x401660 <__main>
   0x000000000040155d <+13>:    lea    rcx,[rip+0x2a9c]        # 0x404000
   0x0000000000401564 <+20>:    call   0x402a90 <puts>
   0x0000000000401569 <+25>:    mov    DWORD PTR [rbp-0x4],0xf93
   0x0000000000401570 <+32>:    mov    DWORD PTR [rbp-0x8],0x2710
   0x0000000000401577 <+39>:    mov    DWORD PTR [rbp-0x20],0x2a6
   0x000000000040157e <+46>:    mov    DWORD PTR [rbp-0x1c],0x37a
   0x0000000000401585 <+53>:    mov    DWORD PTR [rbp-0x18],0x36c
   0x000000000040158c <+60>:    mov    DWORD PTR [rbp-0x14],0x3db
   0x0000000000401593 <+67>:    mov    DWORD PTR [rbp-0x10],0x36c
   0x000000000040159a <+74>:    mov    DWORD PTR [rbp-0xc],0x1c8
   0x00000000004015a1 <+81>:    mov    eax,0x0
   0x00000000004015a6 <+86>:    add    rsp,0x40
   0x00000000004015aa <+90>:    pop    rbp
   0x00000000004015ab <+91>:    ret
End of assembler dump.
(gdb) list
```

below is the gdb's render of the C code used:



```
End of assembler dump.
(gdb) list
        #include <stdio.h>
        #include <stdlib.h>

        int main(){
            printf("Welcome\n");
            int a = 3987;
            int b = 10000;
            int c[6] = {678,890,876,987,876,456};
           // printf("%d", c[0]);
        }
```