

Hw2-3 Taosha Gao

q1.

initialise three variables , give them values of 3, 5, 0

```
eax -> 3,  
Multiply eax with 5 -> 15  
edx -> eax=15  
eax -> 3  
ecx -> eax=3  
ecx -> 1  
eax+ecx -> eax=3  
eax/2 -> eax=1  
edx-eax -> edx=14  
eax -> 14  
[esp+14h] -> 14  
print result
```

q2. print the largest number

initialise ten variables, give them values of 0Ch, 0Fh, 0DDh, 3,1B0h, 36h, 10h, 43h, 0, 0
goto loc_40157F

```
loc_40157F:  
Compare [esp+38h] with 7  
if [esp+38h] equal/lower than 7, jump to loc_401560  
else  
eax -> [esp+3Ch]  
print eax
```

```
loc_401560:  
eax -> [esp+38h]  
eax -> [esp+0*4+18h]  
Compare eax with [esp+3Ch]  
if eax is equal/lower than 0, jump to loc_40157A  
else  
eax -> [esp+38h]  
[esp+3Ch] -> eax
```

```
loc_40157A:  
[esp+38h]+1
```

q3. print the numbers between 100 and 999 such that $(\text{its hundred number})^3 + (\text{its tenth number})^3 + (\text{its unit number})^3 = \text{itself}$
[esp+1Ch] -> 64H
goto loc_4015D6

```
loc_4015D6:  
compare [esp+1Ch] with 3E7h  
If [esp+1Ch] is equal/lower than 3E7h, jump to loc_40151B  
else eax -> 0
```

```
loc_40151B:  
ecx -> [esp+1Ch]  
edx -> 51EB851Fh
```

```

eax -> ecx
edx -> edx*eax
edx/(2^5)
ecx -> eax
eax -> 0
edx -> edx-eax
eax -> edx
[esp+18h] -> eax
edx -> eax*(-64h)
eax -> [esp+1Ch]
ecx -> the address of [edx+eax]
edx -> 66666667h
eax -> ecx
eax -> edx*eax
edx/(2^2)
eax -> ecx
eax -> 1
edx -> edx-eax
[esp+14h] -> eax
ecx -> [esp+1Ch]
edx -> 66666667h
eax -> ecx
edx -> edx*eax
edx/(2^2)
eax -> ecx
eax -> 0
edx -> edx-eax
eax -> edx
eax*2^2
eax -> eax+edx
eax -> eax+eax
ecx -> ecx-eax
eax -> ecx
[esp+10h] -> eax
eax -> [esp+18h]
eax*[esp+18h]
eax*[esp+18h] -> eax*eax*eax
edx -> eax
eax -> [esp+14h]
eax*eax*eax
edx -> edx+eax
eax -> [esp+10h]
eax*eax*eax
eax -> eax+edx
compare eax with [esp+1Ch], if eax isn't equal to [esp+1Ch] then jump to loc_4015D1
else
eax -> [esp+1Ch]
print eax

loc_4015D1:
[esp+1Ch] -> [esp+1Ch]+1

```