Problem 2.8

Translate 0xabcdef12 into decimal.

$= 10*16^7+11*16^6+12*16^5+13*16^4+14*16^3+15*16^2+1*16^1+2*16^0$
$= 2684354560+184549376+12582912+851968+57344+3849+16+2$
$= 2882400018$

Problem 2.16

Provide the type, assembly language instruction, and binary representation of instruction described by the following MIPS fields:
op=0, rs=3, rt=2, rd=3, shamt=0, funct=34
R-type instruction
MIPS code: ***sub $v1,$v0,$v1***

opcode	rs	rt	rd	shamt	funct
000000	00011	00010	00011	00000	100010

Problem 2.18.1

128 registers, 38 bits for each, register bits=log2(128)=7

opcode(6 bits)	rs(7 bits)	rt(7 bits)	rd(7 bits)	shamt(5 bits)	funct(6 bits)

2 additional bits for all the bits. The size of word is 44.

Problem 2.18.2

128 registers, 38 bits for each, register bits=log2(128)=7

opcode(6 bits)	rs(7 bits)	rt(7 bits)	immediate(16 bits)

Size of word is also 44.

Problem 2.18.3

The size of the MIPS assembly program increases since the number of bits increased.
More registers means there is less requirement of register spilling in midway.
At the same time, don't need instruction for the spilling of registers, size of program decresed.

Problem 2.25.1

The appropriate instruction format should be I-type not R-type because the instruction increments the value of PC only if the condition is true

opcode	Source register	Destination register	immediate value

Problem 2.25.2

```
loop:   slt $t3,$zero,$t2
        beq $t3,$zero, Exit
        addi $t2,$t2,-1
```