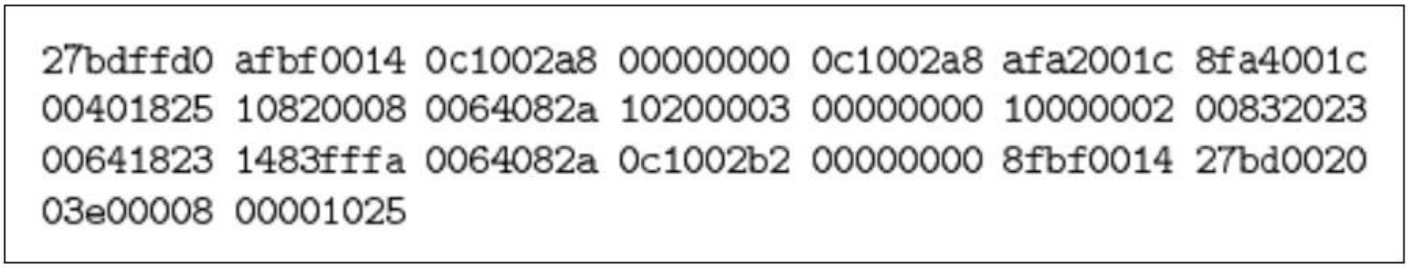
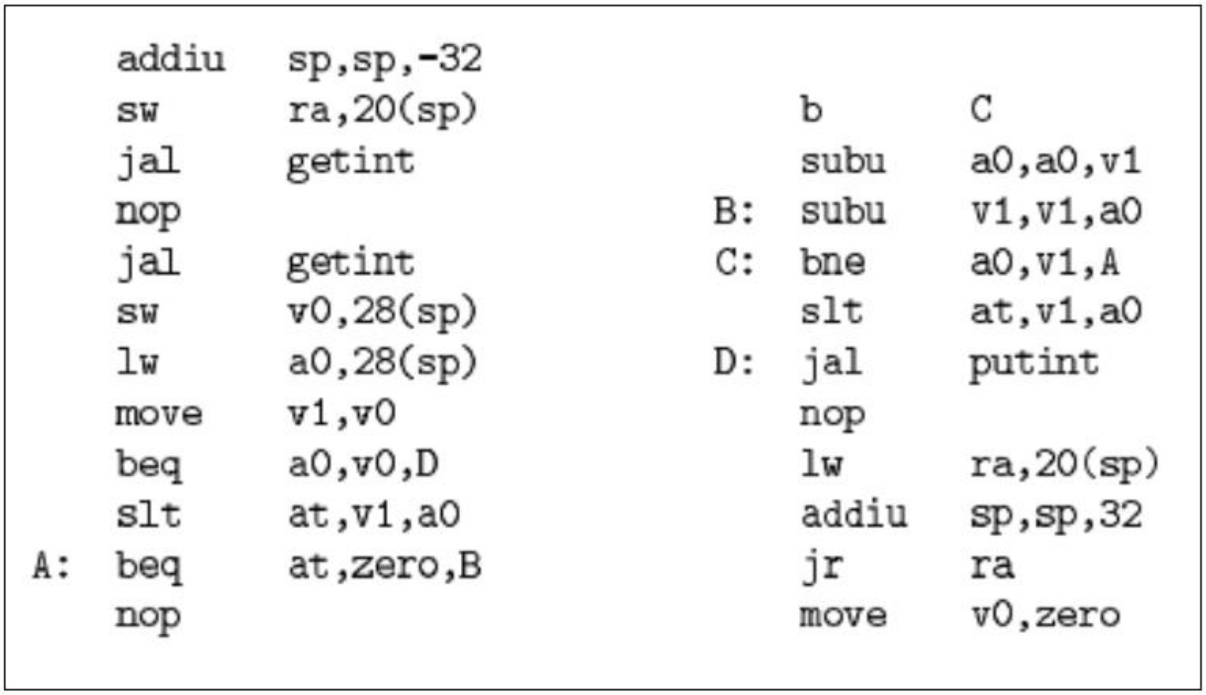
```
//C, C++
int gcd ( int a, int b )
{
  while ( a != b )
  {
   if ( a > b )
     a = a - b;
   else
     b = b - a;
  }
  return a;
}
```

X86 Machine code



```
27bdffd0 afbf0014 0c1002a8 00000000 0c1002a8 afa2001c 8fa4001c
00401825 10820008 0064082a 10200003 00000000 10000002 00832023
00641823 1483fffa 0064082a 0c1002b2 00000000 8fbf0014 27bd0020
03e00008 00001025
```

X86 assembly language code



```
    addiu  sp,sp,-32
    sw     ra,20(sp)                b      C
    jal    getint                   subu   a0,a0,v1
    nop                         B:  subu   v1,v1,a0
    jal    getint               C:  bne    a0,v1,A
    sw     v0,28(sp)                slt    at,v1,a0
    lw     a0,28(sp)            D:  jal    putint
    move   v1,v0                    nop
    beq    a0,v0,D                  lw     ra,20(sp)
    slt    at,v1,a0                 addiu  sp,sp,32
A:  beq    at,zero,B                jr     ra
    nop                             move   v0,zero
```

//Scheme

```
( define gcd
    ( lambda (a b)
        ( cond   ( (= a b) a)
            ( (> a b) (gcd (- a b) b))
            ( else (gcd (- b a) a))
        )

     )

)
```

//Prolog

```
gcd(A, B, G):- A=B, G=A.

gcd(A, B, G):- A>B, C is A-B, gcd(C,B,G).

gcd(A, B, G):- B>A, C is B-A, gcd(A,C,G).
```