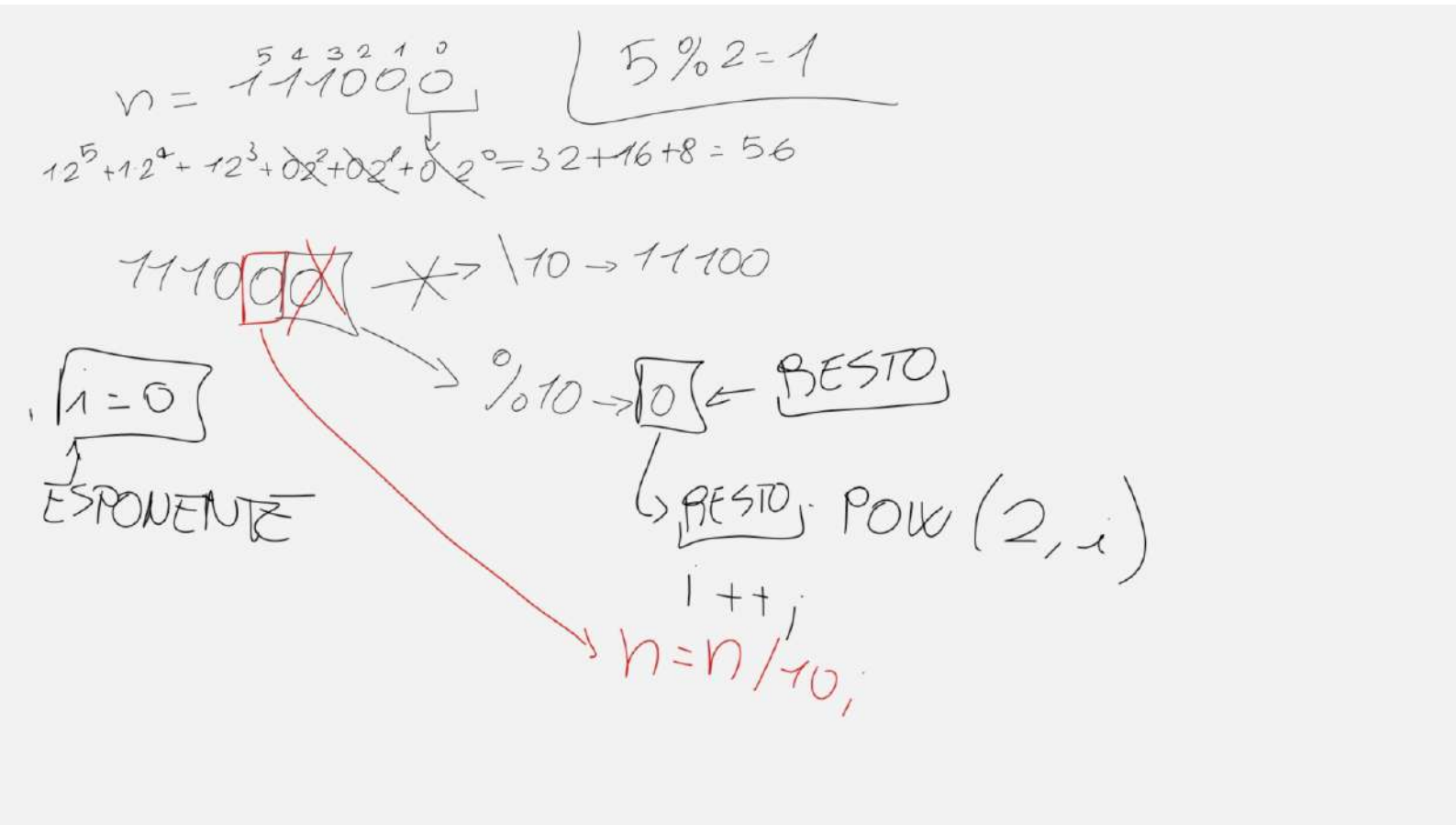
```
void elevaAlQuadrato(int num, int* risultato){
    *risultato = num * num;
    /*
    int QUAD = POW(num, 2);
    *risultato = QUAD;
    */
}
```

COMMENTO

```
int convertiINDecimale(int n){
 int i=0;
 int dec=0;
 int resto;

while(n>0){
resto=n%10;
 dec= dec + resto*(pow(2,i));
     i++;
   n=n/10;
}
        return dec;
         }
```


n = 111000
5 4 3 2 1 0
5 % 2 = 1
1*2^5 + 1*2^4 + 1*2^3 + 0*2^2 + 0*2^1 + 0*2^0 = 32 + 16 + 8 = 56
11100 0 -> /10 -> 11100
% 10 -> 0 <- RESTO
i = 0
EXPONENTE
RESTO * POW (2, i)
i++;
n = n / 10;

WHILE (n>0)

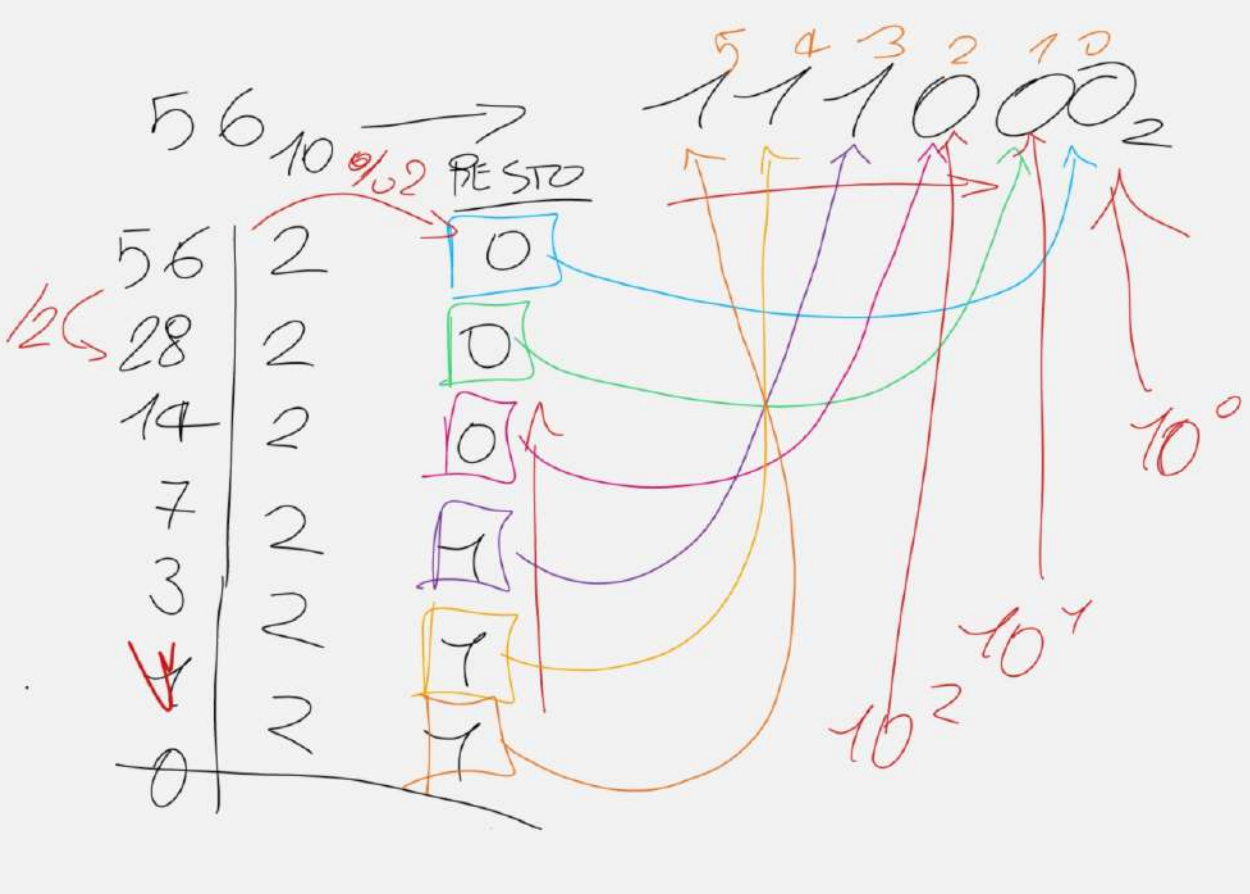
STEP 1 — n % 10 → RESTO
(RESTO = n % 10;)

STEP 2 — DEC = DEC + RESTO * POW(2, i)

STEP 3 — i++

STEP 4 — n = n / 10;

ES 1111000


56 10 → 2
RESTO
56 2 0
28 2 0
14 2 0
7 2 1
3 2 1
1 2 1
0 2 1
12
1 1 1 0 0 0 2
5 4 3 2 1 0
10^0
10^1
10^2

```
int convertInBinario (int n){
    int i=0;
    int binario=0;
    int resto;
    while(n>0){
      resto = n%2;
      binario = binario + resto *(pow(10, i));
      i++;
      n=n/2;
    }
    return binario;
}
```