① Decimal to any base

$(6\ 3\ 4)_{10} = (\ ?\ )_8$

8 | 6 3 4	2	$\times 10^0$ +	→ answer = reverse & add with power of 10
8 | 7 9	7	$\times 10^1$ +	
8 | 9	1	$\times 10^2$ +	
1 1	1	$\times 10^3$	∴ ans = $1\times10^0 + 1\times10^1 + 7\times10^2 + 2\times10^3$.

$(1\ 1\ 7\ 2)_8$

CODE:

```
int ans=0, pow10=1;
while(n>0){
    ans = ans + (n%8) x pow10;
    pow10 = 10 x pow10;
    n = n/8;
}
return ans;
```

Step 1: divident

2 ——↳ divide the number until it becomes zero

3 ——↳ extract remainder, multiply it with multiplier and add to answer.
and update the multiplier

② Any base addition

To add 2 numbers in decimal base,

n1 = 3 2 4

n2 = 7 8 6

$(8+2+1)\%10 = 1_2$ ← $(1+6)/10$

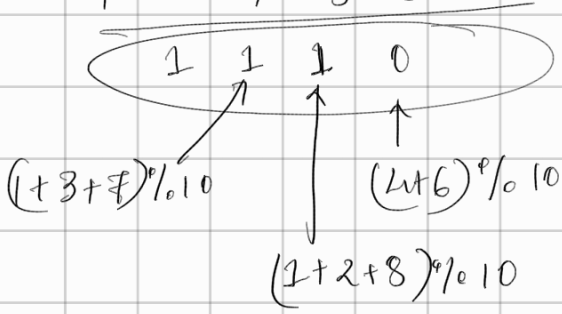
3 2 4

\+ 7 8 6

So, similarly, for the other bases we need to

① add the (digits + carry) = res

② Add (res % base) x pow10 to ans



- It to it's correct place

③ update carry, pow 10 and the 2 numbers n1 and n2.

④ return this value to print.

CODE:

```
int ans = 0, carry = 0, pow10 = 1;
while(n1 > 0 || n2 > 0 || carry > 0) {
    int sum = (n1 % 10) + (n2 % 10) + carry;
    ans += (sum % b) * pow10;
    carry = sum / b;
    pow10 *= 10;
    n1 /= 10;
    n2 /= 10;
}
return ans;
```