Q1)

→ Code :

```
int x = 2018113003%100;
int a= -1*(x);
unsigned int b= (unsigned int) a;
unsigned int c= UINT_MAX-x;
int d = (int)&c;
int p= 65490+x;
short int e =(short int)p
unsigned short f = (unsigned short)a;
printf("%d %u %u %d %hi %hu \n",a,b,c,d,e,f);
```

Result:

The code is for a 32 bit machine

∴ int → 4 bytes

short int → 2 bytes

Evaluation is done right-to-left

∴ x= 2018113003%100

⟹ x= 3, which does not cause any overflow

∴ x=3 (Represented as : 0— 0xxx 0011)

a= -1 * x

= -3, done by two's complement

∴ a= -3 (Represented as : 1→1x27—1101)

Binary of $a = 1 - 1 \times 27 - 1101_2$

∴ (unsigned int) a will not change the binary

∴ $b = 1 - 1 \times 27 - 1101_2$, but it is unsigned
∴ $b = UINT\_MAX - 2$
$= (2^{32} - 1) - 2$
$= 4294967293$

$C = UINT\_MAX - 3$
$= (2^{32} - 1) - 3$
$= 1 - 1 \times 27 - 1100_2$

It is unsigned
∴ $C = 4294967292$

$d = (int) c$
The bit value remains unchanged
∴ $d = 1 - 1 \times 27 - 1100_2$

But it is signed
∴ $d = -(2^s \text{ complement of } 1 - 1 \times 27 - 1100_2)$
$= -(0 - 0 \times 27 - 0100_2)$
$= -4$

$P = 65490 + 3$
$= 654983 \text{ (No overflow)}$

$p = 0 - 0x15 - 1x10 - 0101 01_2$

$\therefore e = (short)p$
$= 1x10 - 0101 01_2$

→ It is signed
$\therefore e = -(1010 10_2 + 1)$
$= -(1010 11_2)$
$= -43$

$f = (unsigned)e$
$f = (unsigned\ short)a$
$= (2^{16} - 1) - 4$
$= 1 - 1x12 - 1101$
$= (2^{16} - 1) - 2$
$= 65533$

$\therefore$ Ans : -3 4294967293 4294967292 -4 -43 65533