

- Evaluate the following unsigned conversions.
 - $(330)_{\text{dec}} = (\text{101001010})_{\text{bin}}$
 - $(01110101000111100011)_{\text{bin}} = (\text{751E3})_{\text{hex}}$
 - $(0110)_{\text{hex}} = (\text{272})_{\text{dec}}$
 - $(1234)_{\text{oct}} = (\text{668})_{\text{dec}}$
 - $(111010100101)_{\text{bin}} = (\text{7245})_{\text{oct}}$
- Evaluate the following binary operations in 8 bit 2s complement arithmetic. Indicate overflows.
 - $10010011 + 10100011 = \text{00110110 (overflow)}$
 - $00000011 - 10101010 = \text{01011001}$
 - $00000101 \times 00001001 = \text{00101101}$
 - $10000010 + 10001010 = \text{00001100 (overflow)}$
 - $01110100 / 00000100 = \text{00011101}$
- Put ✓ mark in ✓ column if the declaration of standard C-type is valid. Put x mark in x column otherwise. Leave both columns blank if you cannot decide. Assume that no new type is defined previously.

✓	x
✓	
	x
✓	
✓	
	x
	x
	x
✓	

✓	x
✓	
✓	
	x
	x
✓	
	x
✓	
	x

- Determine what is printed on the screen when the given code is executed.

```
int main(void){
    printf("Hello %X",16384);
    return 0; }
```

Hello 4000

- Determine the line number of the compiler error in the following function?.

```
int foo(long a, long b){ /* line 1 */
    int a, c=1;          /* line 2 */
    c = a*b-c;           /* line 3 */
    printf("Carpim is %d",c); /* line 4 */
    return 0; }          /* line 5 */
```

2

- Determine what is printed on the screen when the given code is executed.

```
int main(void){ int pi=3.1415, e=2.71828;
    printf("%d",e); printf("%d",pi);
    return 0; }
```

23