# Homework 1

20125071 – Bùi Lê Gia Cát

1. P2.59

```c
void main(){
    int x=0x89ABCDEF;
    int y=0x76543210;

    int ans=y&4294967040; // = y&(2^32-2^8);
    ans+=(x&255);         // = x&(2^8-1);
    printf("The result of 0x765432EF: %i", ans);
}
```

2. P2.60

```c
unsigned replace_byte (unsigned x, int i, unsigned char b){
    unsigned tmp1=pow(2,33)-pow(2,(i+1)*8); //get bits in front
    unsigned tmp2=pow(2,i*8)-1; // get bits in back

    unsigned ans=x&tmp1;
    ans+=(b<<(i*8));    // fill the hole
    ans+=x&tmp2;
    return ans;
}
```

3. P2.61

```c
int testOptionA(int x){
    return !!x;
}

int testOptionB(int x){
    return !!(~x);
}

int testOptionC(int x){
    return !!(x & 0xFF);
}

int testOptionD(int x){
    return !!(~x && (0xFF<<24));
}
```

4. P2.64

```c
int any_odd_one(int x){
    return !!(x & 0xAAAAAAAA);
}
```

5. P2.65

```
int odd_ones(unsigned x){
    x ^= x >> 16;
    x ^= x >> 8;
    x ^= x >> 4;
    x ^= x >> 2;
    x ^= x >> 1;
    return x & 0x1;
}
```

6. P2.66

```
int leftmost_one(unsigned x){
    x |= x >> 16;
    x |= x >> 8;
    x |= x >> 4;
    x |= x >> 2;
    x |= x >> 1;

    return (x >> 1) + ( x && 1);
}
```

7. P2.68

```
int lower_one_mask(int n){
    return (2<<(n-1))-1;
}
```

8. P2.77

```
int multipleBy17(int x){
    return (x<<4)+x;        // <=> 16x+x
}

int multipleByMinus7(int x){
    return -(x<<3)+x;       // <=> -8x+x;
}

int multipleBy60(int x){
    return (x<<6)-(x<<2);   // <=> 64x-4x;
}

int multipleByMinus112(int x){
    return -(x<<7)+(x<<4);  // <=> -128x + 16x;
}
```

9. P2.79

```
int mul3div4(int x){
    return x-(x>>2);
}
```

10. P2.81

```
unsigned bitPatternTypeA(int k){
    int w=sizeof(int)*8;
    unsigned ans=1<<(w-k);
    ans--;
    ans<<=k;
    return ans;
}

unsigned bitPatternTypeB(int k, int j){
    unsigned ans = 1<<(k);
    ans--;
    ans<<=j;
    return ans;
}
```