HW 2. Embedded C programming

Prob 2-1. (70 points total, 5 points each) Read the code below and answer the questions for `C[0]` to `'C[13]`:

```c
int main(void) {
    uint16_t A[4] = {0x5555, 0x6666, 0x9999, 0xAAAA};
    uint16_t B[4] = {0x6666, 0x7777, 0xAAAA, 0xBBBB};
    uint16_t C[20];
    uint16_t Mask = 7;        // a mask for 3 bits
    uint16_t NUM_of_bits_to_shift = 4;
    uint16_t VALUE_to_assign = 5;
    uint16_t *pInt;

    //printf("The address of array C: %x\n", C);
    printf("Please determine the values in Hexadecimal of C:\n");
    pInt = &A[1];
    C[0] = A[0] & B[0];
    C[1] = A[1] | B[1];
    C[2] = A[2] && B[2];
    C[3] = A[3] ^ B[3];
    C[4] = 15 % 4;
    C[5] = Mask << NUM_of_bits_to_shift;
    C[6] = ~(Mask << NUM_of_bits_to_shift);
    C[7] = A[0] & ~(Mask << NUM_of_bits_to_shift);
    C[8] = (VALUE_to_assign << NUM_of_bits_to_shift);
    C[9] = A[1] | (VALUE_to_assign << NUM_of_bits_to_shift);
    C[10] = *(pInt + 2);
    C[11] = *pInt++;
    C[12] = (*pInt)++;
    C[13] = *pInt;
```

Prob 2-2 (10 points, 5 each) Convert the following hexadecimal numbers to 16-bit binary number. (Example `0xFF = 0b0000_0000_1111_1111`) a. `0x1F`, b. `0x1234`

Prob 2-3 (10 points) Consider a 12-bit ADC (analog to digital converter), which converts a voltage from 0 to 5 V to a digital number between 0 and the maximum unsigned number that a 12-bit system can express. What is that maximum number?

Prob 2-4 (20 points) Determine the exact address of the IDR for GPIO port B according to the code on pages 9 and 10 of the class notes.