1. What are the GPIO control registers that the lab mentions? Briefly describe each of their functions.
GPIO port mode register (GPIOx_MODER) - It is used to configure the I/O mode.
GPIO port output type register (GPIOx_OTYPER)- It is used to configure I/O output type.
GPIO port output speed register (GPIOx_OSPEEDR)- It is used to configure the I/O output speed.
GPIO port pull-up/pull-down register (GPIOx_PUPDR)- it is used to configure the I/O pull-up or pull-down
GPIO port input data register (GPIOx_IDR)- These are read-only. It stores the value of corresponding I/O value.
GPIO port output data register (GPIOx_ODR)- It is used to drive output low or high by setting appropriate bits.
GPIO port bit set/reset register (GPIOx_BSRR)- This register is write-only. The reason is that this register is a shortcut to set and clear bits quickly in the output register. The lower half of this register sets bits in the output, and the upper half clears/resets them.

GPIO port configuration lock register (GPIOx_LCKR)- This register is used to lock the configuration of the port bits when a correct write sequence is applied to bit 16 (LCKK).

GPIO alternate function low/high registers (GPIOx_AFRL/GPIOx_AFRH)-it is used to configure the alternate functions of the I/Os

GPIO port bit reset register (GPIOx_BRR)- It is a write only. A read to these bit returns value of 0x0000

2. What values would you want to write to the bits controlling a pin in the GPIOx_MODER register in order to set it to analog mode?

Setting ‘11’ to any register of GPIOx_MODER
GPIOx->MODER |= (0x3 << (5 * 2)); (x= Ato F)

3. Examine the bit descriptions in GPIOx_BSRR register: which bit would you want to set to clear the fourth bit in the ODR?
Set 19th bit of BSRR to 1 to clear 4th of ODR

4. Perform the following bitwise operations:
• 0xAD | 0xC7 = 10101101 | 11000111= 11101111=239
• 0xAD & 0xC7 = 10000101 = 133
• 0xAD & ~(0xC7) = 00101000 = 40
• 0xAD ^0xC7 = 01101010= 106

5. How would you clear the 5th and 6th bits in a register while leaving the other's alone?

We can use bitwise AND operation to clear 5th and 6th bits keeping every other bits unchanged.

6. What is the maximum speed the STM32F072R8 GPIO pins can handle in the lowest speed setting?
• Use the chip datasheet: lab section 1.4.1 gives a hint to the location. You'll want to search the I/O AC characteristics table. You will also need to view the OSPEEDR settings to find the bit pattern indicating the slowest speed.
Max frequency = 2MHZ

7. What RCC register would you manipulate to enable the following peripherals: (use the comments next to the bit defines for better peripheral descriptions)
• TIM1 (TIMER1) = RCC->APB2ENR |= RCC_APB2ENR_TIM1EN; ('1' timer clock enabled - Bit 11)

• DMA1 = RCC->AHBENR |= RCC_AHBENR_DMA1EN; ('1' DMA clock enabled- Bit - 0)

• I2C1 = RCC->APB1ENR |= RCC_APB1ENR_I2C1EN; ('1' I2C1 clock enabled - Bit 21)