

Лабораторная работа №6
Битовые (поразрядные) операции.

1. Постановка задачи:

Запустите программу и объясните результат.

Результат:

```
#include<stdio.h>

//#include<conio.h>

void main()

{

int i,a,b; // а и b вместе занимают 8 байт/64 бита?

/* ----- */

printf("\nВведите два целых числа: "); scanf("%d %d",&a,&b);

printf("\n");

/* ----- */

printf("Операция 'инверсия битов' в числе a: 0x%x=%d\n", ~a,~a); /* %x — целое число типа int
со знаком в шестнадцатеричной системе счисления, %d — целое число типа int со знаком в
десятичной системе счисления */

printf("Операция 'инверсия битов' в числе b: 0x%x=%d\n", ~b,~b); /* каждый бит числа
получает обратное своему значению (0/1) соответственно */

printf("\n");

/* ----- */

printf("Операция 'сдвиг <<2' в числе a : 0x%x=%d\n", a<<2,a<<2); /* 01000100<<2 = 000010000*/

printf("Операция 'сдвиг >>2' в числе a : 0x%x=%d\n", a>>2,a>>2); /* 01000100>>2 = 00010001*/

printf("\n");

/* ----- */

printf("Операция 'битовое И' a&b : 0x%x=%d\n", a&b,a&b); /* поразрядное сравнение битов
числа a и b логическим И */

printf("Операция 'битовое ИЛИ' a|b : 0x%x=%d\n", a|b,a|b); /* поразрядное сравнение битов
числа a и b логическим ИЛИ */

printf("Операция 'битовое исключающее ИЛИ ': 0x%x=%d\n", a^b,a^b);

/* поразрядное сравнение битов числа a и b исключающим ИЛИ*/

printf("\n");

}
```

2.Постановка задачи: Выполните программу и объясните результат.

Допишите код.

Результат:

```
#include<stdio.h>

//#include<conio.h>

void main()

{

int i,a,b;

/* ----- */

printf("\nВведите два целых числа: "); scanf("%d %d",&a,&b);

printf("\n");

printf("Номера и значения битов старшего байта числа a:\n");

for (i=15;i>=8;i--) /*представление байта в 16значном виде, работа с первыми 8 битами*/

printf("(%d,%d) ",i,(a>>i)&0x01); /* побитовое сравнение с единицей */

printf("\n\n");

/* ----- */

printf("Номера и значения битов младшего байта числа a:\n");

for (i=7;i>=0;i--) /*представление байта в 16значном виде, работа с вторыми 8 битами*/

printf("(%d,%d) ",i,(a>>i)&0x01);

printf("\n\n");

/* ----- */

printf("Биты числа b в прямом порядке : ");

for (i=15;i>=0;i--)

printf("%d",(b>>i)&0x01);

printf("\n");

/* ----- */

printf("Биты числа b в обратном порядке: "); /* Допишите код, ссылаясь на

пример: «Биты числа b в прямом порядке» */

for (i=0;i<=15;i++)

printf("%d",(b>>i)&0x01);

printf("\n\n");

/* ----- */
```

```
//getch();
```

```
}
```

3.Постановка задачи: Объясните, как работает программа.

```
#include<stdio.h>
```

```
//#include<conio.h>
```

```
void main()
```

```
{
```

```
unsigned int n;
```

```
printf("Введите натуральное число: "); scanf("%u",&n);
```

```
if (n&0x0001)
```

```
printf("Число %u является нечётным.\n",n);
```

```
else printf("Число %u является чётным.\n",n);
```

```
//getch();
```

```
}
```

Результат: на вход принимается положительное целое число, если его последний бит = 1 то оно нечетное, иначе четное.

4.Постановка задачи: Запустите код и объясните результат программы «Использование операции "сдвиг вправо" для вывода на экран двоичного представления данного натурального числа»

Заполните комментарии к программе.

Результат:

```
#include<stdio.h>
```

```
//#include<conio.h>
```

```
void main()
```

```
{
```

```
long int a; /* Исходное натуральное число:
```

```
если a<255, то для хранения числа требуется _____ памяти;
```

```
если a<65535, то для хранения числа требуется _____ памяти;
```

```
если a<2147483647, то для хранения числа требуется _____ памяти */
```

```
int i,n; // Параметр цикла
```

```
printf("Введите натуральное число : "); scanf("%lu",&a);
```

```
n=(a>255)?(a>65535)?31:15:7; /* if a>255
```

```

        if a>65535

        n=31

        else n=15

        else n = 7*/

printf("Его двоичное представление: ");

for (i=n;i>=0;i--)

{

printf("%d",(a>>i)&0x1); // побитовое сравнение числа a с единицей

if (i%8==0) //для простановки пробелов между каждыми 8 битами

printf(" ");

}

printf("\n");

//getch();

}

```

5. Постановка задачи: Выполните программу и объясните результат КАЖДОЙ операции.

Результат:

```

#include<stdio.h>

//#include<conio.h>

void main()

{

int y,x,z,k;

x=03; y=02; z=01;

k=x|y&z; printf("1. %d\n",k); /* 01&10=00,00|11=11=3 */

k=x|y&~z; printf("2. %d\n",k); /* ~01=10, 10&10=10, 11|10=11=3 */

k=x^y&~z; printf("3. %d\n",k); /* ~01=10, 10&10=10, 11^10=01=1 */

k=x&y&&z; printf("4. %d\n",k); /* в документации microsoft написано, что && соответствует &,
но судя по результату с небулевыми переменными он работает по другому*/

x=1; y=-1;

k=!x|x; printf("5. %d\n",k); /* !00|01=01 */

k=-x|x; printf("6. %d\n",k); /* -01|01=-01 */

k=x^x; printf("7. %d\n",k); /* 01^01=00 */

x<<=3; printf("8. %d\n",x); /* 00000001<<3=00001000 */

y<<=3; printf("9. %d\n",y); /* -00000001<<3=-00001000 */

```

```
y>=3; printf("10. %d\n\n",y); /*-00001000>>3=-00000001*/  
//getch();  
}
```