

Лабораторная работа №1

Представление данных в ЭВМ

Выполнили: Шувалова Виктория

Сахно Андрей

Решетников Егор

МИЭТ

Москва

2020

Задние 1-5

```
#include <iostream>

#include "stdint.h"

#include "math.h"

#include <bitset>

#include <limits>


template<typename T>
void print(T x)
{
    std::bitset<16> xBin(x);

    std::bitset<16> xBinUnsigned((unsigned int)x);

    std::cout << std::dec << "dec: " << x << " hex: " << std::hex << x << " binary: " << xBin <<
std::endl;

    std::cout << "UNSIGNED\n";

    std::cout << std::dec << "dec: " << (uint16_t)x << " hex: " << std::hex << x << " binary: " <<
xBinUnsigned << std::endl << std::endl;

}


template<typename T>
void print_hex_dec(T x)
{
    std::cout << "hex: " << std::hex << x << " dec: " << std::dec << x << std::endl;

}


int main()
```

```

{
// 1

std::cout
<<"_____ \n";

std::cout <<" | \n";
std::cout <<" | \n";

std::cout <<"Задание 1. Изучите, как интерпретируется одна и та же область памяти, если \n";
std::cout <<"она рассматривается как знаковое или беззнаковое число, а также как одно и то \n";
std::cout <<"же число записывается в различных системах счисления. \n";
std::cout <<"Необходимо сравнить: \n";
std::cout <<"а) беззнаковую интерпретацию переменной в шестнадцатеричной форме; \n";
std::cout <<"б) беззнаковую интерпретацию в десятичной форме; \n";
std::cout <<"в) знаковую интерпретацию в десятичной форме. \n";
std::cout <<"Для этого определите и запишите в отчёт десятичное, двоичное (16 бит) и \n";
std::cout <<"шестнадцатеричное представления шестнадцатитрибитных чисел X и Y, \n";
std::cout <<"а также беззнаковую интерпретацию этого представления в десятичном виде. \n \n";

int16_t x = -2;
int16_t y = pow(2, 14) + 1;

std::cout << "X = -2: \n";
print(x);
std::cout << "y = 2^14+1 \n";
print(y);

// 2

std::cout
<<"_____ \n";

std::cout <<" | \n";
std::cout <<" | \n";

std::cout <<"Задание 2. Найдите и выпишите в отчёт минимальное и максимальное 16- \n";

```

```

std::cout << "битное число со знаком и без знака в формах представления (а), (б), (в) и в двоичной\
n";

std::cout << "форме (4 числа, каждое из которых представлено в 4 формах).\n\n";

//std::cout << "2" << std::endl;

int16_t min = std::numeric_limits<int16_t>::min();
int16_t max = std::numeric_limits<int16_t>::max();

std::cout << "for max 16-bit signed value\n";
print(max);

std::cout << "for min 16-bit signed value\n";
print(min);

// 3

std::cout
<< "_____ \n";

std::cout << "_____ |\n";

std::cout << "_____ |\n";

std::cout << "Задание 3. Разработайте программу на языке C++, выполняющую над
беззнаковыми \n";

std::cout << "шестнадцатитбитными целыми числами следующие поразрядные операции\n";

std::cout << "(результат должен печататься в десятичной и шестнадцатеричной формах):\n";

std::cout << "– бинарные  $x \wedge y$  (конъюнкция),  $x \vee y$  (дизъюнкция),\n";

std::cout << " $x \oplus y$  (сложение по модулю два);\n";

std::cout << "– унарные  $\neg x$  (отрицание),\n";

std::cout << "neg(x) (дополнение до двух,  $x + \text{neg}(x) = 2^{\text{(разрядность } x\text{)}}$ );\n";

std::cout << "–  $x \ll y$  (логический сдвиг влево),  $x \gg y$  (логический сдвиг вправо),\n\n";

//std::cout << "\n3\n";

```

```

uint16_t x_unsigned = 0x9211;
uint16_t y_unsigned = 0x0004;

std::cout << "disjunction\n";
print_hex_dec(x_unsigned|y_unsigned);
std::cout << "\nconjunction\n";
print_hex_dec(x_unsigned&y_unsigned);
std::cout << "\nxor\n";
print_hex_dec(x_unsigned^y_unsigned);
std::cout << "\nx = 0x9211 negation\n";
print_hex_dec(~x_unsigned);
std::cout << "\ny = 0x0004 negation\n";
print_hex_dec(~y_unsigned);
std::cout << "\nleft shift: 0x9211 << 0x0004\n";
print_hex_dec(x_unsigned << y_unsigned);
std::cout << "\nright shift: 0x9211 >> 0x0004\n";
print_hex_dec(x_unsigned >> y_unsigned);

// 4
std::cout
<<"
std::cout <<"
std::cout <<"
std::cout <<"Задание 4. Измените в программе из задания 3 тип переменных на знаковый.\n";
std::cout <<"Объясните изменение (или неизменность) результата.\n\n";

//std::cout << "\n4\n";

int16_t x_signed = 0x9211;

```

```

int16_t y_signed = 0x0004;

std::cout << "disjunction\n";

print_hex_dec(x_signed|y_signed);

std::cout << "\nconjunction\n";

print_hex_dec(x_signed&y_signed);

std::cout << "\nxor\n";

print_hex_dec(x_signed^y_signed);

std::cout << "\nx = 0x9211 negation\n";

print_hex_dec(~x_signed);

std::cout << "\ny = 0x0004 negation\n";

print_hex_dec(~y_signed);

std::cout << "\nleft shift: 0x9211 << 0x0004\n";

print_hex_dec(x_signed << y_signed);

std::cout << "\nright shift: 0x9211 >> 0x0004\n";

print_hex_dec(x_signed >> y_signed);

// 5

std::cout
<<"
std::cout <<"
std::cout <<"
std::cout <<"Задание 5. Бонус (+1 балл). Разработайте программу на языке C++ \n";
std::cout <<"(или дополните программу из задания 3), которая расширяет шестнадцатитбитное\
n";

std::cout <<"представление числа x до тридцатидвухбитного, рассматривая числа как\n";

std::cout <<"– знаковые (signed);\n";

std::cout <<"– беззнаковые (unsigned).\n\n";

//std::cout << "\n5\n";

int16_t x_closed = 0xFF00;

```

```

int32_t x_expanded = x_closed;

std::cout << "0xFF00 in 16 bt\n";

print_hex_dec(x_closed);

std::cout << "0xFF00 in 32 bt\n";

print_hex_dec(x_expanded);


uint16_t x_closed_unsigned = 0xFF00;

uint32_t x_expanded_unsigned = 0xFF00;

std::cout << "unsigned 0xFF00 in 16 bt\n";

print_hex_dec(x_closed_unsigned);

std::cout << "unsigned 0xFF00 in 32 bt\n";

print_hex_dec(x_expanded_unsigned);


return 0;
}

```

Поделим вывод соответственно заданиям.

Задание №1

```

int16_t x = -2;

int16_t y = pow(2, 14) + 1;


std::cout << "X = -2:\n";

print(x);

std::cout << "y = 2^14+1\n";

print(y);

```

Задание 1. Изучите, как интерпретируется одна и та же область памяти, если она рассматривается как знаковое или беззнаковое число, а также как одно и то же число записывается в различных системах счисления.

Необходимо сравнить:

- а) беззнаковую интерпретацию переменной в шестнадцатеричной форме;
- б) беззнаковую интерпретацию в десятичной форме;
- в) знаковую интерпретацию в десятичной форме.

Для этого определите и запишите в отчёт десятичное, двоичное (16 бит) и шестнадцатеричное представления шестнадцатитрибитных чисел x и y , а также беззнаковую интерпретацию этого представления в десятичном виде.

$x = -2$:

dec: -2 hex: fffe binary: 1111111111111110

UNSIGNED

dec: 65534 hex: fffe binary: 1111111111111110

$y = 2^{14} + 1$

dec: 16385 hex: 4001 binary: 0100000000000001

UNSIGNED

dec: 16385 hex: 4001 binary: 0100000000000001

Видно, что десятичная интерпретация изменилась. Т.к. первый бит теперь не используется как знаковый, произошло переполнение.

Задание №2

```
int16_t min = std::numeric_limits<int16_t>::min();
int16_t max = std::numeric_limits<int16_t>::max();
uint16_t u_min = std::numeric_limits<uint16_t>::min();
uint16_t u_max = std::numeric_limits<uint16_t>::max();

std::cout << "for max 16-bit signed value\n";
print(max);

std::cout << "for min 16-bit signed value\n";
print(min);

std::cout << "for max 16-bit unsigned value\n";
print(u_max);

std::cout << "for min 16-bit unsigned value\n";
print(u_min);
```


Задание 2. Найдите и выпишите в отчёт минимальное и максимальное 16-битное число со знаком и без знака в формах представления (а), (б), (в) и в двоичной форме (4 числа, каждое из которых представлено в 4 формах).

```
for max 16-bit signed value
dec: 32767 hex: 7fff binary: 0111111111111111
UNSIGNED
dec: 32767 hex: 7fff binary: 0111111111111111

for min 16-bit signed value
dec: -32768 hex: 8000 binary: 1000000000000000
UNSIGNED
dec: 32768 hex: 8000 binary: 1000000000000000

for max 16-bit unsigned value
dec: 65535 hex: ffff binary: 1111111111111111
UNSIGNED
dec: 65535 hex: ffff binary: 1111111111111111

for min 16-bit unsigned value
dec: 0 hex: 0 binary: 0000000000000000
UNSIGNED
dec: 0 hex: 0 binary: 0000000000000000
```

Задание №3

```
uint16_t x_unsigned = 0x9211;
uint16_t y_unsigned = 0x0004;
std::cout << "disjunction\n";
print_hex_dec(x_unsigned|y_unsigned);
std::cout << "\nconjunction\n";
print_hex_dec(x_unsigned&y_unsigned);
std::cout << "\nxor\n";
print_hex_dec(x_unsigned^y_unsigned);
std::cout << "\nx = 0x9211 negation\n";
print_hex_dec(~x_unsigned);
std::cout << "\ny = 0x0004 negation\n";
print_hex_dec(~y_unsigned);
std::cout << "\nleft shift: 0x9211 << 0x0004\n";
```

```
print_hex_dec(x_unsigned << y_unsigned);

std::cout << "\nright shift: 0x9211 >> 0x0004\n";

print_hex_dec(x_unsigned >> y_unsigned);
```

Задание 3. Разработайте программу на языке C++, выполняющую над беззнаковыми шестнадцатитривиными целыми числами следующие поразрядные операции (результат должен печататься в десятичной и шестнадцатеричной формах):

- бинарные $x \wedge y$ (конъюнкция), $x \vee y$ (дизъюнкция),
- $x \oplus y$ (сложение по модулю два);
- унарные $\neg x$ (отрицание),
- $\text{neg}(x)$ (дополнение до двух, $x + \text{neg}(x) = 2^{\text{(разрядность } x \text{)}}$),
- $x \ll y$ (логический сдвиг влево), $x \gg y$ (логический сдвиг вправо)..

```
disjunction
hex: 9215 dec: 37397
```

```
conjunction
hex: 0 dec: 0
```

```
xor
hex: 9215 dec: 37397
```

```
x = 0x9211 negation
hex: ffff6dee dec: -37394
```

```
y = 0x0004 negation
hex: ffffffff dec: -5
```

```
left shift: 0x9211 << 0x0004
hex: 92110 dec: 598288
```

```
right shift: 0x9211 >> 0x0004
hex: 921 dec: 2337
```

Задание №4

```
int16_t x_signed = 0x9211;

int16_t y_signed = 0x0004;

std::cout << "disjunction\n";

print_hex_dec(x_signed|y_signed);

std::cout << "\nconjunction\n";

print_hex_dec(x_signed&y_signed);

std::cout << "\nxor\n";

print_hex_dec(x_signed^y_signed);

std::cout << "\nx = 0x9211 negation\n";
```

```
print_hex_dec(~x_signed);  
std::cout << "\ny = 0x0004 negation\n";  
print_hex_dec(~y_signed);  
std::cout << "\nleft shift: 0x9211 << 0x0004\n";  
print_hex_dec(x_signed << y_signed);  
std::cout << "\nright shift: 0x9211 >> 0x0004\n";  
print_hex_dec(x_signed >> y_signed);
```

Задание 4. Измените в программе из задания 3 тип переменных на знаковый. Объясните изменение (или неизменность) результата.

```
disjunction  
hex: ffff9215 dec: -28139
```

```
conjunction  
hex: 0 dec: 0
```

```
xor  
hex: ffff9215 dec: -28139
```

```
x = 0x9211 negation  
hex: 6dee dec: 28142
```

```
y = 0x0004 negation  
hex: ffffffff dec: -5
```

```
left shift: 0x9211 << 0x0004  
hex: fff92110 dec: -450288
```

```
right shift: 0x9211 >> 0x0004  
hex: fffff921 dec: -1759
```

Видны различия. Но по десятичной форме можно понять их природу: все дело в том, как битовая запись интерпретируется в число. В результате мы стали считать первый бит знаковым и получать соответствующее значение.

Задание №5

```
int16_t x_closed = 0xFF00;
int32_t x_expanded = x_closed;
std::cout << "0xFF00 in 16 bt\n";
print_hex_dec(x_closed);
std::cout << "0xFF00 in 32 bt\n";
print_hex_dec(x_expanded);
uint16_t x_closed_unsigned = 0xFF00;
uint32_t x_expanded_unsigned = 0xFF00;
std::cout << "unsigned 0xFF00 in 16 bt\n";
print_hex_dec(x_closed_unsigned);
std::cout << "unsigned 0xFF00 in 32 bt\n";
print_hex_dec(x_expanded_unsigned);
```

Задание 5. Бонус (+1 балл). Разработайте программу на языке C++ (или дополните программу из задания 3), которая расширяет шестнадцатитбитное представление числа x до тридцатидвухбитного, рассматривая числа как

- знаковые (signed);
- беззнаковые (unsigned).

```
0xFF00 in 16 bt
hex: ff00 dec: -256
0xFF00 in 32 bt
hex: ffffffff00 dec: -256
unsigned 0xFF00 in 16 bt
hex: ff00 dec: 65280
unsigned 0xFF00 in 32 bt
hex: ff00 dec: 65280
```

Задание №6

```
#include <iostream>

#include <bitset>

void print_four_chars(char some_str[])
{
    for (int i = 0; i < 4; i++)
    {
        std::cout << some_str[i] << ": " << (int)some_str[i] << std::endl;
    }
}

int main()
{
    std::cout << "Задание 6. Определите и выпишите в отчёт, как хранятся в памяти компьютера:\n";
    std::cout << "– целое число 0x12345678; по результату исследования определите порядок\n";
    std::cout << "следования байтов в словах для вашего процессора:\n";
    std::cout << "– Little-Endian (от младшего к старшему, порядок Intel)\n";
    std::cout << "– Big-Endian (от старшего к младшему, порядок Motorola)\n";
    std::cout << "– строки 'abcd' и 'абвг' (массив из char);\n";
    std::cout << "– «широкие» строки L'abcd' и L'абвг' (массив из wchar_t).\n\n";

    // 6

    int32_t x = 0x12345678;;
    std::bitset<32> x_bin(x);

    std::cout << "0x12345678\n";
    std::cout << "dec: " << std::dec << x << std::endl;
    std::cout << "binary: " << x_bin << std::endl;
```

```
char eng_string[] = "abcd";
char rus_string[] = "aᄁBr";

std::cout << "\n abcd:\n";
for (int i = 0; i < 8; i++)
{
    std::cout << (int)rus_string[i] << " ";
}

std::cout << "\naᄁBr\n";
for (int i = 0; i < 4; i++)
{
    std::cout << (int)eng_string[i] << " ";
}

std::cout << std::endl;

wchar_t expanded_eng_string[] = L"abcd";
wchar_t expanded_rus_string[] = L"aᄁBr";

std::cout << "\nL\"abcd\"\n";

for (int i = 0; i < 4; i++)
{
    std::cout << (int)expanded_eng_string[i] << " ";
}

std::cout << "\nL\"aᄁBr\"";
```

```

for (int i = 0; i < 4; i++)
{
    std::cout << (int)expanded_rus_string[i] << " ";
}

return 0;
}

```

Задание 6. Определите и выпишите в отчёт, как хранятся в памяти компьютера:

- целое число 0x12345678; по результату исследования определите порядок следования байтов в словах для вашего процессора;
- Little-Endian (от младшего к старшему, порядок Intel)
- Big-Endian (от старшего к младшему, порядок Motorola)
- строки 'abcd' и 'абвг' (массив из char);
- «широкие» строки L'abcd' и L'абвг' (массив из wchar_t).

```

0x12345678
dec: 305419896
binary: 00010010001101000101011001111000

```

```

abcd:
-48 -80 -48 -79 -48 -78 -48 -77
абвг
97 98 99 100

```

```

L"abcd"
97 98 99 100
L"абвг"1072 1073 1074 1075 [1] + Done
(base) d@ludak@d@ludak: ~/lab6 $ ^C

```

"/usr/bin/gdb" -

Переведя двоичную запись в десятичную (через степени 2), получим что байты следуют от старшего к младшему.

Задание №7

```
#include <iostream>

#include <cstdint>

#include <string>

template<typename T>

void print_size(std::string type_name)

{

    std::cout << type_name << ": " << sizeof(T) << std::endl;

}

int main()

{

    std::cout << "Задание 7. При помощи оператора sizeof выясните, сколько байтов\n";

    std::cout << "занимают переменные следующих типов: char, bool, wchar_t, short, int,\n";

    std::cout << "long, long long, float, double, long double, size_t, ptrdiff_t,\n";

    std::cout << "void*. Результаты оформите в отчёте в виде таблицы, указывая для каждого типа\n";

    std::cout << "его назначение.\n";

    std::cout << "Для выполнения единообразных действий над переменными различных типов\n";

    std::cout << "используются макросы препроцессора C или шаблоны C++.\n";

    std::cout << "Проверьте, соответствуют ли размеры типов современному стандарту C++.\n\n";

    print_size<char>("char");

    print_size<bool>("bool");

    print_size<wchar_t>("wchar_t");

    print_size<short>("short");

    print_size<int>("int");

    print_size<long>("long");

    print_size<long long>("long long");

    print_size<float>("float");

    print_size<double>("double");
```



```
print_size<long double>("long double");

print_size<size_t>("size_t");

print_size<ptrdiff_t>("ptrdiff_t");

print_size<void*>("void*");
```

```
return 0;
```

```
}
```

Задание 7. При помощи оператора `sizeof` выясните, сколько байтов занимают переменные следующих типов: `char`, `bool`, `wchar_t`, `short`, `int`, `long`, `long long`, `float`, `double`, `long double`, `size_t`, `ptrdiff_t`, `void*`. Результаты оформите в отчёте в виде таблицы, указывая для каждого типа его назначение.

Для выполнения единообразных действий над переменными различных типов используются макросы препроцессора C или шаблоны C++.

Проверьте, соответствуют ли размеры типов современному стандарту C++.

```
char: 1
bool: 1
wchar_t: 4
short: 2
int: 4
long: 8
long long: 8
float: 4
double: 8
long double: 16
size_t: 8
ptrdiff_t: 8
void*: 8
```

Тип данных	Результат (ubuntu x64)	Win 32	Win 64	Ubuntu x32	Стандарт c++
char	1	1	1	1	1
bool	1	1	1	1	1
wchar_t	4	2	2	4	2
short	2	2	2	2	2
int	4	4	4	4	2
long	8	4	4	4	4
Long long	8	8	8	8	8
float	4	4	4	4	4
double	8	8	8	8	4
Long double	16	8	8	12	8
size_t	8	4	8	4	2
ptrdiff_t	8	4	8	4	4
void*	8	4	8	4	4