

федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский университет
«Московский институт электронной техники»

Направление подготовки 09.03.04 «Программная инженерия»
Дисциплина «Электротехника»

Лабораторная РАБОТА №7
Ассемблерные вставки в программах на C++. Флаги и условия

Работу выполнили студенты группы ПИН-24 Баранов Д.А. и Демочкина А.В.
Работу проверил ассистент Института СПИНТех Фомин Р.А.

Цель работы: научиться анализировать флаги x86 и использовать условные команды.

Задание 1.

Вариант 2

Вычислите сумму двух целых чисел $z = x + y$. Определите корректность (отсутствие/наличие переполнения) результата, если трактовать его как

2	беззнаковый
---	-------------

```
#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    unsigned int x, y, z;
    cin >> x;    //x1 = 4294967295, y1 = 1 - переполнение
    cin >> y;    //x1 = 4294967295, y1 = 0 - нет переполнения - max(unsigned int)
    bool bo;
    asm (
        "add    %%ebx, %%eax\n\t"
        : "=a"(z), "=ccc"(bo)    //1 - переполнение, 0 - нет беззнаковое переполнения
        : [X]"a"(x), [Y]"b"(y)
        : "cc"
    );
    cout << "z = " << z << endl;
    cout << bo;
    return 0;
}
```

```
#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    unsigned int x, y, z;
    cin >> x;    //x1 = 4294967295, y1 = 1 - переполнение
    cin >> y;    //x1 = 4294967295, y1 = 0 - нет переполнения - max(unsigned int)
    bool bo;
    asm (
        "add    %%ebx, %%eax\n\t"
        : "=a"(z), "=ccc"(bo)    //1 - переполнение, 0 - нет беззнаковое переполнения
        : [X]"a"(x), [Y]"b"(y)
        : "cc"
    );
    cout << "z = " << z << endl;
    cout << bo;
    return 0;
}
```

Нет переполнения:

4294967295 0

Program returned: 0

Program stdout

Ассемблер:

z = 4294967295

0

Переполнение:

4294967295 1

Program returned: 0

Program stdout

Ассемблер:

z = 0

1

СС: с - значение регистра флага состояния CF (CF = 1 - беззнаковое переполнение)

Задание 2.

Вариант 2

Вычислите для заданного целого беззнакового x :

2	$z = (x \geq 2)$
---	------------------

```
#include <cstdio>
#include <iostream>
using namespace std;
```

```
int main() {
    cout << "Ассемблер: " << endl;
    unsigned int x;
    bool z;
    cin >> x;
    asm (
        "cmp    %%eax, %%ebx\n\t" // сравнение для целого беззнакового числа
        : "=@ccbe"(z) // be - больше либо равно (eax >= ebx). 1 - больше, 2 - меньше
        : [X]"a"(x), [Y]"b"(2)
        : "cc"
    );
    cout << "z = " << z << endl;
    return 0;
}
```

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    unsigned int x;
    bool z;
    cin >> x;
    asm (
        "cmp    %%eax, %%ebx\n\t" // сравнение для целого беззнакового числа
        : "=@ccbe"(z) //be - больше либо равно (eax >= ebx). 1 - больше, 2 - меньше
        : [X]"a"(x), [Y]"b"(2)
        : "cc"
    );
    cout << "z = " << z << endl;
    return 0;
}

```

1	2	23434
Program returned: 0	Program returned: 0	Program returned: 0
Program stdout	Program stdout	Program stdout
Ассемблер:	Ассемблер:	Ассемблер:
z = 0	z = 1	z = 1

CC: be - значение регистра флагов состояния CF и ZF (CF = 1 и ZF = 1 - первое число больше либо равно второму как беззнаковое)

Задание 3.

Реализуйте Л7.32 для целого знакового *x*.

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    int x;
    bool z;
    cin >> x;
    asm (
        "cmp    %%eax, %%ebx\n\t" // сравнение для целого знакового числа
        : "=@ccle"(z) //le - больше либо равно (eax >= ebx). 1 - больше, 2 - меньше
        : [X]"a"(x), [Y]"b"(2)
        : "cc"
    );
    cout << "z = " << z << endl;
    return 0;
}

```

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    int x;
    bool z;
    cin >> x;
    asm (
        "cmp    %%eax, %%ebx\n\t" // сравнение для целого знакового числа
        : "=ccle"(z) //le - больше либо равно (eax >= ebx). 1 - больше, 2 - меньше
        : [X]"a"(x), [Y]"b"(2)
        : "cc"
    );
    cout << "z = " << z << endl;
    return 0;
}

```

-34	1	2	3423423
Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout
Ассемблер: z = 0	Ассемблер: z = 0	Ассемблер: z = 1	Ассемблер: z = 1

CC: le - значение регистра флагов состояния SF, OF и ZF (SF != OF и ZF = 1 - первое число больше либо равно второму как знаковое)

Задание 4.

Реализуйте Л7.32 для x с плавающей запятой двойной точности (*double*), используя AVX-команду сравнения *vcomisd* (или её SSE-аналог *comisd*).

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    double x;
    bool z;
    cin >> x;
    asm (
        "vcomisd  %[X], %[Y]\n\t" // сравнение для числа двойной точности (double)
        : "=ccbe"(z) //be - больше либо равно (x >= y). 1 - больше, 0 - меньше
        : [X]"x"(x), [Y]"x"(2.0)
        : "cc"
    );
    cout << "z = " << z << endl;
    return 0;
}

```

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    double x;
    bool z;
    cin >> x;
    asm (
        "vcomisd    %[X], %[Y]\n\t" // сравнение для числа двойной точности (double)
        : "=ccbe"(z) //be - больше либо равно (x >= y). 1 - больше, 0 - меньше
        : [X] "x"(x), [Y] "x"(2.0)
        : "cc"
    );
    cout << "z = " << z << endl;
    return 0;
}

```

1.999999	-34534.345	2	22323.324
Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout
Ассемблер: z = 0	Ассемблер: z = 0	Ассемблер: z = 1	Ассемблер: z = 1

sd - двойная точность (double)

СС: be - значение регистра флагов состояния CF и ZF (CF = 1 и ZF = 1 - первое число больше либо равно второму как число двойной точности)

<code>vcomi(sd ss) sxm, sx</code>	$ZF = \begin{cases} 1, & \text{если } sx - sxm = 0 \text{ или } \text{qnan} \\ 0, & \text{иначе} \end{cases}$ $CF = \begin{cases} 1, & \text{если } sx - sxm < 0 \text{ или } \text{qnan} \\ 0, & \text{иначе} \end{cases}$ $PF = \begin{cases} 1, & \text{несравнимы } (sx - sxm - \text{qnan}) \\ 0, & \text{сравнимы} \end{cases}$
-----------------------------------	---

Задание 5.

Реализуйте Л7.32 для x с плавающей запятой двойной точности, используя AVX-команду сравнения *vcmps*d (или её SSE-аналог) и битовые операции.

```
#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    double x, y;
    bool z;
    cin >> x;
    cin >> y;
    asm("vcmpsд $0b10, %%xmm0, %%xmm1, %%xmm0\n\t" // $0b10 - непосредственная
        // константа, указывающая
        // какое сравнение нужно
        // произвести(>=). Сравнение
        // xmm0 и xmm1 и запись
        // результата в xmm0
        "vandpd %[A], %%xmm0, %%xmm0\n\t" // Логическое "И" - избавление от
        // результата 11...111.
        // 1 - x больше y. 0
        // - x меньше y
        "vmovsd %%xmm0, %[Z]\n\t"
        : [Z] "=m"(z)
        : [X] "x"(x), [Y] "x"(y), [A] "x"($0b1)
        : "memory");
    cout << "z = " << hex << z << endl;
    return 0;
}
```

```
#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    double x, y;
    bool z;
    cin >> x;
    cin >> y;
    asm("vcmpsд $0b10, %%xmm0, %%xmm1, %%xmm0\n\t" // $0b10 - непосредственная
        // константа, указывающая
        // какое сравнение нужно
        // произвести(>=). Сравнение
        // xmm0 и xmm1 и запись
        // результата в xmm0
        "vandpd %[A], %%xmm0, %%xmm0\n\t" // Логическое "И" - избавление от
        // результата 11...111.
        // 1 - x больше y. 0
        // - x меньше y
        "vmovsd %%xmm0, %[Z]\n\t"
        : [Z] "=m"(z)
        : [X] "x"(x), [Y] "x"(y), [A] "x"($0b1)
        : "memory");
    cout << "z = " << hex << z << endl;
    return 0;
}
```

```

: [Z] "=m"(z)
: [X] "x"(x), [Y] "y"(y), [A] "x"(0b1)
: "memory");
cout << "z = " << hex << z << endl;
return 0;
}

```

1.9999 2	-2434.2 2	2 2	666.666 2
Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout
Ассемблер: z = 0	Ассемблер: z = 0	Ассемблер: z = 1	Ассемблер: z = 1

1 × double: vcmpsd vcomisd		1 × float: vcmpss vcomiss	
vcmp (sd ss) imm8, sxm, sx, dx		$dx[0] = \begin{cases} 111...111, & \text{если } sxm[0] \text{ imm8 } sx[0] \\ 000...000, & \text{иначе} \end{cases}$	
<i>imm8</i>	операция	<i>imm8</i>	операция
0b 000X X000	$sxm[0] = sx[0]$	0b 000X X100	$sxm[0] \neq sx[0]$
0b 000X X001	$sxm[0] > sx[0]$	0b 000X X101	$sxm[0] \not\geq sx[0]$
0b 000X X010	$sxm[0] \geq sx[0]$	0b 000X X110	$sxm[0] \not\neq sx[0]$
0b 000X X011	несравнимы	0b 000X X111	сравнимы
X — любой бит (0 или 1).		Флаги: <i>flags</i> не изменяется, <i>mxcsr</i> : DE IE	

Задание 6.

Реализуйте Л7.32 для x с плавающей запятой двойной точности, используя FPU-команды сравнения $f[u]comi[p]$.

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    double x;
    const double y = 2.0;
    cin >> x;
    bool b = 1;
    asm(
        "fldl %[x]\n\t" // в стеке: x // l - суффикс размера (double)
        "fldl %[y]\n\t" // в стеке: y, x
        "fucomi    %%st(1), %%st(0)" //сравнение: st(1) с st(0)
        :[B]="@ccbe"(b) //st(1) >= st(0) = x >= y
        :[X]"m"(x), [Y]"m"(y)
        : "cc"
    );
    cout << b;
    return 0;
}

```

```

#include <cstdio>
#include <iostream>
using namespace std;

```



```
int main() {
    double x;
    const double y = 2.0;
    cin >> x;
    bool b = 1;
    asm(
        "fldl %[x]\n\t" // в стеке: x    // l - суффикс размера (double)
        "fldl %[y]\n\t" // в стеке: y, x
        "fucomi    %%st(1), %%st(0)" //сравнение: st(1) с st(0)
        : [B] "=@ccbe" (b) //st(1) >= st(0) = x >= y
        : [x] "m" (x), [y] "m" (y)
        : "cc"
    );
    cout << b;
    return 0;
}
```

-234.3	1.99	2.0	24.232
Program returned: 0	Program returned: 0	Program returned: 0	Program returned: 0
Program stdout	Program stdout	Program stdout	Program stdout
0	0	1	1

CC*	Флаги	Результат ρ сл/выч	sub/cmp src, dest	and/test src, dest	fucomi %st(i), %st(0)
na/be	$CF = 1$ $ZF = 1$	$dest - src \leq 0$	инанков	$dest \& src = 0$	$st(0) - src \leq 0$
Суффиксы размера команд FPU с <i>stmem</i> , <i>dmem</i>		операнд	вещественный (f)	целый (fi)	
		s или без суффикса	single (32)	short (16)	
		l	double (64)	long (32)	
		t	ten bytes (80)		

Задание 7.

AVX:

```
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    float x;
    bool z;
    cin >> x;
    asm (
        "vcvtsd2ss  %[Y], %[Y], %[Y]\n\t" //преобразуем число в формат float
        "vcomiss    %%xmm0, %%xmm1\n\t" //be - больше либо равно (x >= 2). 1 - больше, 0 - меньше
        : "=@ccbe" (z)
        : [X] "x" (x), [Y] "x" (2.0)
        : "cc"
    );
    cout << "z = " << z << endl;
    return 0;
}
```

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    cout << "Ассемблер: " << endl;
    float x;
    bool z;
    cin >> x;
    asm (
        "vcvtsd2ss  %[Y], %[Y], %[Y]\n\t" //преобразуем число в формат float
        "vcomiss    %%xmm0, %%xmm1\n\t" //be - больше либо равно (x >= 2). 1 - больше, 0 - меньше
        : "=@ccbe"(z)
        : [X]"x"(x), [Y]"x"(2.0)
        : "cc"
    );
    cout << "z = " << z << endl;
    return 0;
}

```

1.99999	2.0	-234	344.43
Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout
Ассемблер: z = 0	Ассемблер: z = 1	Ассемблер: z = 0	Ассемблер: z = 1

ss -одинарная точность (float)

CC: be - значение регистра флагов состояния CF и ZF (CF = 1 и ZF = 1 - первое число больше либо равно второму как число одинарной точности)

<code>vcomi(sd ss) sxm, sx</code>	$ZF = \begin{cases} 1, & \text{если } sx - sxm = 0 \text{ или } \text{qnan} \\ 0, & \text{иначе} \end{cases}$ $CF = \begin{cases} 1, & \text{если } sx - sxm < 0 \text{ или } \text{qnan} \\ 0, & \text{иначе} \end{cases}$ $PF = \begin{cases} 1, & \text{несравнимы } (sx - sxm - \text{qnan}) \\ 0, & \text{сравнимы} \end{cases}$
-----------------------------------	---

FPU:

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    float x;
    const float y = 2.0;
    cin >> x;
    bool b = 1;
    asm(
        "flds %[x]\n\t" // в стеке: x   // s / отсутствие - суффикс размера (float)
        "flds %[y]\n\t" // в стеке: y, x
        "fucomi    %%st(1), %%st(0)" //сравнение: st(1) с st(0)
        : [B]"= @ccbe"(b) //st(1) >= st(0) = x >= y
        : [X]"m"(x), [Y]"m"(y)
        : "cc"
    );
    cout << b;
    return 0;
}

```

```

#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    float x;
    const float y = 2.0;
    cin >> x;
    bool b = 1;
    asm(
        "flds %[x]\n\t" // в стеке: x    // s / отсутствие - суффикс размера (float)
        "flds %[y]\n\t" // в стеке: y, x
        "fucomi    %%st(1), %%st(0)" //сравнение: st(1) с st(0)
        :[B] "=@ccbe"(b) //st(1) >= st(0) = x >= y
        :[x] "m"(x), [y] "m"(y)
        : "cc"
    );
    cout << b;
    return 0;
}

```

-14	1.999	2.0	777.777
Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout
0	0	1	1

СС*	Флаги	Результат ρ сл/выч	sub/cmp src, dest	and/test src, dest	fucomi %st(i), %st(0)
na/be	$CF = 1$ $ZF = 1$	$dest - src \leq 0$	волок	$dest \& src = 0$	$st(0) - src \leq 0$

**Суффиксы
размера**
 команд FPU
 с *stmem*, *dmem*

операнд	вещественный (f)	целый (fi)
s или без суффикса	single (32)	short (16)
l	double (64)	long (32)
t	ten bytes (80)	

Задание 8.

Реализуйте Л7.32 для x с плавающей запятой двойной расширенной точности (*long double*).

```
#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    long double x;
    const long double y = 2.0;
    cin >> x;
    bool b = 1;
    asm(
        "fldt %[x]\n\t" // в стеке: x    // t - суффикс размера (long double)
        "fldt %[y]\n\t" // в стеке: y, x
        "fucomi    %%st(1), %%st(0)" //сравнение: st(1) с st(0)
        :[B]"=@ccbe"(b) //st(1) >= st(0) = x >= y
        :[x]"m"(x), [y]"m"(y)
        : "cc"
    );
    cout << b;
    return 0;
}
```

```
#include <cstdio>
#include <iostream>
using namespace std;

int main() {
    long double x;
    const long double y = 2.0;
    cin >> x;
    bool b = 1;
    asm(
        "fldt %[x]\n\t" // в стеке: x    // t - суффикс размера (long double)
        "fldt %[y]\n\t" // в стеке: y, x
        "fucomi    %%st(1), %%st(0)" //сравнение: st(1) с st(0)
        :[B]"=@ccbe"(b) //st(1) >= st(0) = x >= y
        :[x]"m"(x), [y]"m"(y)
        : "cc"
    );
    cout << b;
    return 0;
}
```

-242.1	1.9	2	12
Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout	Program returned: 0 Program stdout
0	0	1	1

СС*	Флаги	Результат ρ сл/выч	sub/cmp src, dest	and/test src, dest	fucomi %st(i), %st(0)
na/be	$CF = 1$ $ZF = 1$	$dest - src \leq 0$	ВОЗМОЖНО	$dest \& src = 0$	$st(0) - src \leq 0$

**Суффиксы
размера
команд FPU
с *smem, dmem***

операнд	вещественный (f)	целый (fi)
s или без суффикса	single (32)	short (16)
l	double (64)	long (32)
t	ten bytes (80)	