例题

1. 在x86-64机器上运行下列代码，输出是？

|  |
| --- |
| **unsigned** A **=** **0x11112222;**  **unsigned** B **=** **0x33336666;**  **void** **\***x **=** **(void\*)&**A**;**  **void** **\***y **=** **2** **+** **(void\*)&**B**;**  **unsigned** **short** P **=** **\*(unsigned** **short\*)**x**;**  **unsigned** **short** Q **=** **\*(unsigned** **short\*)**y**;**  printf**(**"0x%04x"**,** P **+** Q**);** |

**0x5555**

1. 在x86-64机器上，对任意的整型x和y值，ux和uy分别为其转化成无符号数的值，则下面等价的是（不成立的给出反例）：

|  |  |  |  |
| --- | --- | --- | --- |
| x>y | ux > uy | **N** | **x = 0, y = -1** |
| (x > 0) || (x < ux) | 1 | **N** | **x <= 0** |
| x^y^x^y^x | x | **Y** |  |
| ((x >> 1) << 1) <= x | 1 | **Y** |  |
| ((x / 2) \* 2) <= x | 1 | **N** | **-1** |
| x ^ y ^ (~x) - y | y ^ x ^ (~y) - x | **Y** | **优先级** |
| (x == 1) && (ux – 2 < 2) | (x == 1) && ((!!ux) - 2) < 2 | **N** | **!的返回值是有符号数** |

1. 在采用小端法存储机器上运行下面的代码，输出的结果将会是？

(int，unsigned 为 32 位长，short 为 16 位长，0~9 的 ASCII 码分别是 0x30~0x39)

|  |
| --- |
| **char** **\***s **=** "2018"**;**  **int** **\***p1 **=** **(int** **\*)**s**;**  **short** s1 **=** **(\***p1**)>>12;**  **unsigned** u1 **=** **(unsigned)** s1**;**  printf**(**"0x%x\n"**,**u1**);** |

1. 0x00002303 B) 0x00032303 C) 0xffff8313 D) 0x00008313

**C**

4. 假设浮点数格式A为1符号+3阶码+4小数，浮点数格式B为1符号+4阶码+3小数。回答下列问题。

* 格式A中有多少个二进制表示对应于正无穷大?

**只有一个01110000**

* 考虑能精确表示的实数的最大绝对值。A比B大还是比B小，还是两者一样?

**A：01101111表示15.5；B：01110111表示了240；因此B大。**

* 考虑能精确表示的实数的最小非零绝对值。A比B大还是比B小，还是两者一样?

**A：00000001表示1/64； B：00000001表示1/512，因此A大。**

* 考虑能精确表示的实数的个数。A比B多还是比B少，还是两者一样?

**A能精确表达的非负数个数为7\*16=112，B能精确表达的非负数个数为 15\*8=120，因此B能精确表达的实数更多。**

**-> 实际上，A格式表示NaN的数比B格式多。**

5. 判断以下说法的正确性。

|  |  |
| --- | --- |
| Description | True? |
| 对于任意的单精度浮点数 a 和 b ，如果 a > b， 那么a + 1 > b | Y N |
| 对于任意的单精度浮点数 a 和 b，如果 a > b， 那么a + b > b + b | Y N |
| 对于任意的双精度浮点数 d，如果 d < 0，那么 d\*d>0 | Y N |
| 对于任意的双精度浮点数 d，如果 d < 0，那么 d\*2<0 | Y N |
| 对于任意的双精度浮点数 d，d == d | Y N |
| 将 float 转换成 int 时，既有可能造成舍入，又 有可能造成溢出 | Y N |

**(1)正确**

**(2)取a=INF b=FLT\_MAX  
(3) d取最大的非规格化负数**

**(4)正确**

**(5) NaN != NaN  
(6) 正确**

6. 在遵守 IEEE 754 标准的机器上声明 double f, g, h 以及函数原型 int foo()；在给定条件下，判断下列表达式是否恒真：

|  |  |  |
| --- | --- | --- |
| Condition | Description | Always True? |
| f > g | f + 1 > g + 1 | Y N |
| f > g && g > 1 | f – 1 > g - 1 | Y N |
| f = foo();  g = foo();  h = foo(); | (f + g) + h == f + (g + h) | Y N |
| f != 0.0 | f \* f / f == f | Y N |
| f != 0.0 | f / f \* f == f | Y N |

**N: f = 0.0, g为最大负数**

**N: f = 2^53+6, g = 2^53+4**

**Y: double可以完全表示int**

**N: inf**

**N: inf**

练习

1. 变量 x 的值为 0x01234567，地址 &x 为 0x100；则该变量的值在 x86 和 Sun 机器内存 中的存储排列顺序正确的是 ( )

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 选项 | 机器类型 | 地址 | | | |
| 0x100 | 0x101 | 0x102 | 0x103 |
| A | x86 | 67 | 45 | 23 | 01 |
| Sun | 01 | 23 | 45 | 67 |
| B | x86 | 76 | 54 | 32 | 10 |
| Sun | 01 | 23 | 45 | 67 |
| C | x86 | 01 | 23 | 45 | 67 |
| Sun | 67 | 45 | 23 | 01 |
| D | x86 | 01 | 23 | 45 | 67 |
| Sun | 01 | 23 | 45 | 67 |

**A**

2. 对于下面的每一个表达式， 请选择以下选项中的一个或多个(即“不定项”)，使得该表达式恒成立，如果没有满足条件的选项则选 E。

A. < B. > C. == D. != E. none

题目中出现的变量定义如下:

int x, y;

unsigned ux = x;

1) 如果 x > 0, 则 x + 1 0

2) 如果 x > y, 则 ux y

3) 如果 ((x << 31) >> 31) < 0, 则 x & 1 0

4) 如果 ((unsigned char)x >> 1) < 64， 则 (char)x 0

5) xˆyˆ(~x) - y yˆxˆ(~y)-x

6) (((!!ux)) << 31) >> 31) (((!!x) << 31) >> 31)

**1) D 只能保证x+1!=0，可能大于0，也可能小于0（溢出）**

**2) D x>y可能是因为正负，但转成unsigned之后负数可能很大**

**3) B,D (x<<31)>>31实际上提取了x最低一位，其小于0说明最低一位是1，于是&1为1>0**

**4) E 显然可能为0也可能不为0，只能选E**

**5) C 首先我们要关注运算顺序，减法优先于位异或，于是表达式实际上是x^y^((~x)-y)**

**与x^y^((~y)-x)，而-x=~x+1，于是~x-y=~y-x=-x-y-1，于是为==**

**6) C x为0是自然的，x不为0时!返回值为有符号型。**

3. 生成任意 int 类型的 x，然后将它转换为 unsigned 类型：

int x = random();

unsigned ux = (unsigned) x;

对于以下每个 C 语言表达式，判断它的值是否恒为 1。如果是，圈出 Y； 否则圈出 N。

|  |  |
| --- | --- |
| 表达式 | 恒为 1？ |
| (x >= 0) || (x < ux) | Y N |
| ( (x>>1) << 1) <= x | Y N |

**N: e.g. x = -1.**

**Y: x>>1 rounds toward minus infinity.**

4. 下面说法正确的是：

A．数 0 的反码表示是唯一的

B．数 0 的补码表示不是唯一的

C．1000, 1111, 1110, 1111, 1100, 0000, 0000, 0000 表示唯一的整数是 0x8FEFC000

D．1000, 1111, 1110, 1111, 1100, 0000, 0000, 0000 如果是单精度浮点表示，则表示的是-(1.110111111)2\*231-127

**D**

5. 下列说法正确的是：

A. 在 64 位机器上有 int 型变量 a 和 char 型变量 b, 则 sizeof(b) -

sizeof(a) < 0 为真

B. 若 x 为整型，则(x >> 1) << 1 <= x 为真

C. 假设 a, b 为 32 位浮点数，ia, ib 是分别与 a, b 位级表示相同的 32 位有

符号整型数，则 a < b 等价于 ia < ib

D. 双精度浮点数所能表示的规格化数的阶码范围为 [-127, 126]

**B**