寄存器

1、%rip中存放的内容为（   将要执行的指令的位置  ）。

2、f函数返回值存放在哪个寄存器中？ %rax

3、f(x)中的x是通过哪个寄存器来进行的数值传递？ %rdi

# 浮点数谜题

一、对于以下的C语言表达式, 完成以下两者之一:

* 确认其对于所有值均成立
* 如不成立，解释为什么不成立
* **f == -(-f);**
* **成立**
* **2/3 == 2/3.0**
* **不成立，因为整数运算中2/3等于0，在浮点数运算中2/3.0不等于0。**
* **d < 0.0 ⇒ ((d\*2) < 0.0)**
* **成立**
* **d > f ⇒ -f > -d**
* **成立**
* **d \* d >= 0.0**
* **成立**

# 进制转换

1. 十进制整数15215所对应的二进制值为（ 11101101101111  ）

2、考虑以下C语言代码：

    short  si = –2048;

    int  i = si;

执行上述程序段后，i的机器数表示为（  B   ）。

A.FFFF DFFCH           B.FFFF F800H

C.0000 9FFCH           D.0000 DFFCH

传送指令

1、mov (%rax), %rbx 的功能是（ C  ）。

（A）将%rax的值送到以%rbx的值为地址的内存区域中

  (B) 将%rbx的值送到以%rax的值为地址的内存区域中

  (C) 将以%rax的值为地址的内存区域中的值送到%rbx中

  (D) 将以%rbx的值为地址的内存区域中的值送到%rax中

2、假定全局数组a的声明为char \*a[6]，a的首地址为0x401010，i 在rcx中，现要将a[i]取到%rax器中，则所用的汇编指令是（ A  ）。

1. mov 0x401010 ( , %rcx, 8), %rax
2. mov (0x401010, % rcx, 1), % rax
3. mov 0x401010 ( , % rcx), %rax
4. mov (0x401010, % rcx), %rax

# 数组

1. 关于两个N\*N矩阵的加和运算，比较哪一种元素索引顺序的运算速度最快,并说明理由？

(1). 从上到下索引，以列遍历全部元素。

(2). 从左到右索引，以行遍历全部元素。

(2)因为计算机硬件cache的结构,从内存中取逻辑位置较相邻的地址取值,使程序较大可能缓存命中,减少运行时间

2. 假定静态short型二维数组b和指针数组pb的声明如下：

static short b[2][4]={ {2, 9}, {3, 1, -6, 2 }};

static short \*pb[2]={b[0], b[1]};

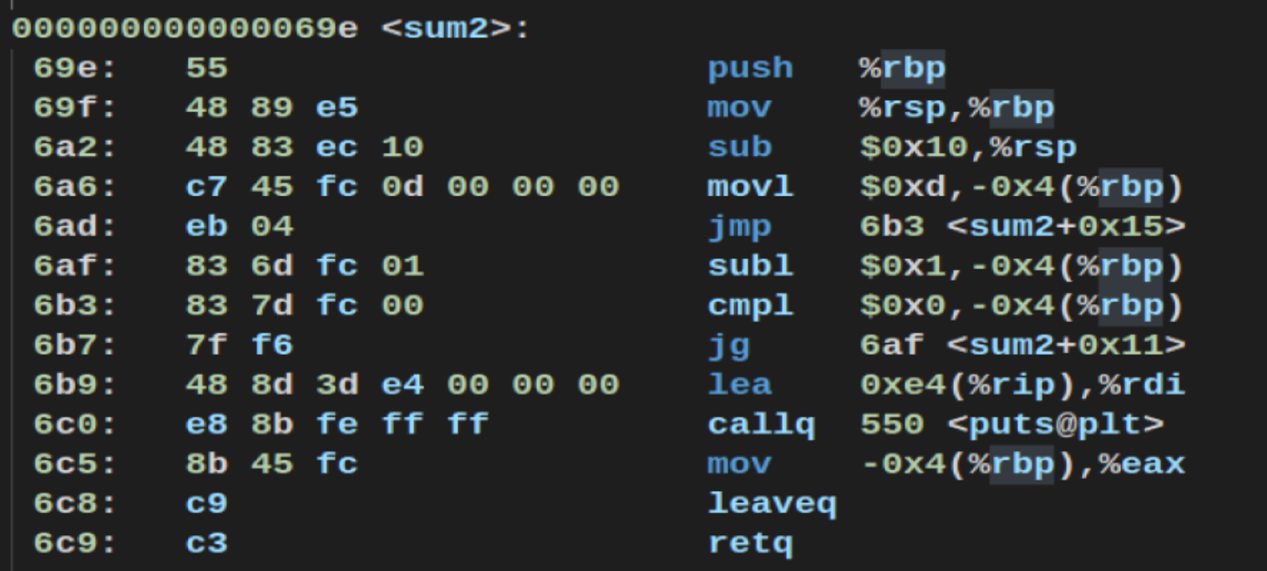
若b的首地址为0x601920，则&pb[1]的值是（ A   ）。

A.0x601938    B. 0x601934

C. 0x601930    D. 0x601928

# 汇编改写C

如下汇编，写出其对应的C程序，局部变量命名没有要求：



int sum2(){

int v1 = 0xd;

while(v1 > 0){

v1--;

}

puts(0x7a4); // 0x6c0+0xe4

return v1;

}

指令

1. 指令pushq %rbp的功能是什么 A

A、R[rsp] <—— R[rsp] - 8， M[R[rsp]] <—— R[rsp]

B、R[rsp] <—— R[rsp] + 8， M[R[rsp]] <—— R[rsp]

C、M[R[rsp]] <—— R[rsp] ，R[rsp] <—— R[rsp] - 8

D、M[R[rsp]] <—— R[rsp] ，R[rsp] <—— R[rsp] + 8

2. 指令popq %rbp的功能是什么 D

A、R[rsp] <—— R[rsp] - 8， M[R[rsp]] <—— R[rsp]

B、R[rsp] <—— R[rsp] + 8， M[R[rsp]] <—— R[rsp]

C、R[rsp]<—— M[R[rsp]]  ，R[rsp] <—— R[rsp] - 8

D、R[rsp] <—— M[R[rsp]]  ，R[rsp] <—— R[rsp] + 8

3. 指令movl 8(%rbp)，%edx的功能是什么？ B

A、M[R[rbp]+8] <—— R[edx]

B、R[edx] <—— M[R[rbp]+8]

C、R[edx] <—— R[rbp]+8

D、R[rbp]+8 <—— R[edx]

数据类型转换

判断下列表达式的正确性：

(A).（unsigned）-1 > -2

(B). 2147483647 > 2147483648U

(C).2147483647U > -2147483647-1

(D). 0U > -1

(E).2147483647 > (int)2147483648U

(F). -1 < 0

T F F F T T