1.

Void doit(void \*args){

Int \*num = (int\*)args;

For(int i = 0; i < M; i++){

Hash\_insert(hashtb, num[i]);

}

2. void init(counter \*c){

c->value = 0;

}

void increment(counter\_t \*c){

fetch\_and\_add(&(c->value), 1);

}

void decrement(counter\_t \*c){

fetch\_and\_add(&(c->value), -1);

}

int get(counter\_t \*c){

return c->value;

}

3. a的初值为1，b的初值为0

void \*t1(void \*arg){

[1]: P(&a);

[2]: V(&b); V(&b);

}

void \*t2(void \*arg){

[1]: P(&b);

[2]:if (b == 0) V(&a);

}

4.

Int CAS(unsigned long \*dst, unsigned long oldVal, unsigned long newVal){

Unsigned char ret;

\_\_asm\_\_ \_\_volatile\_\_(

“ lock: cmpxchgq %2,%1\n”

“ sete %0\n”

: “=q”(ret), “=m”(\*dst)

:”r”(newVal), “m”(\*dst), “a”(oldVal)

:”memory”);

If(ret) return 1;

Return 0;

}

Void List\_Insert(){

Node\_t \*new\_node = malloc(sizeof(node\_t));

If(new\_node == NULL){

perror(“malloc”);

return;

}

new\_node->key = key;

int tmp;

do{

new\_node ->next = L->head;

tmp = CAS(&(L->head), new\_node->next, new\_node);

}while(tmp == 0);

}