**第三章课后习题**

1180300220-崔涵

**3.59**

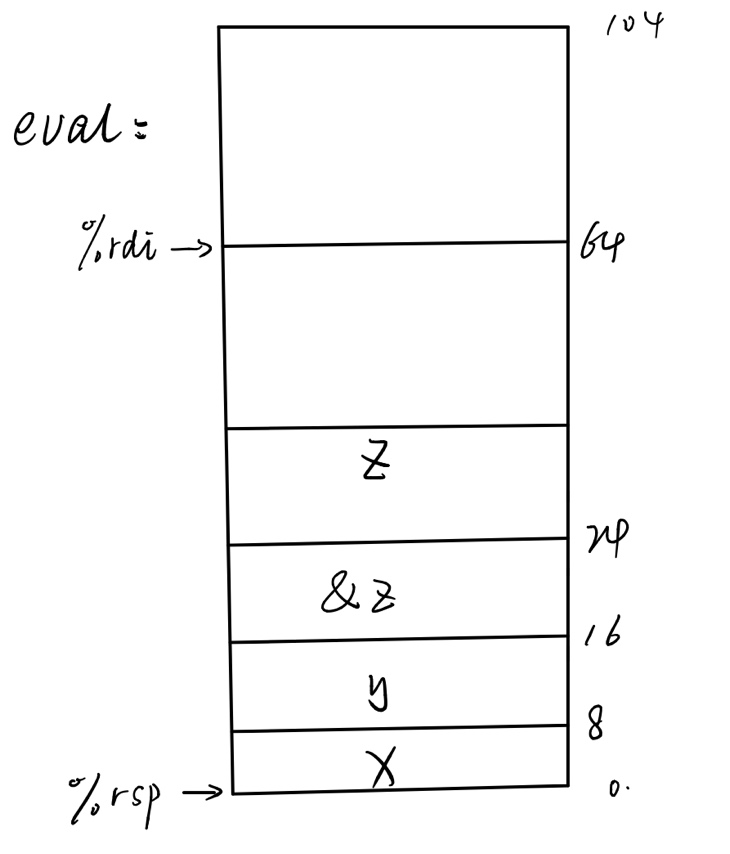
1. // \*dest in %rdi, x in %rsi, y in %rdx
2. // x\*y = (2^64 \* x\_63 + x\_0) \* (2^64 \* y\_63 + y\_0)
3. = 2^64 \* x\_63 \* 2^64 \* y\_63（溢出） + x\_0 \* y\_63 \* 2^64 + x\_63 \* y\_0 \* 2^64 + x\_0 \* y\_0
4. store\_prod:
5. movq    %rdx, %rax //将y存入%rax
6. cqto    //将y的符号位扩展至rdx中
7. movq    %rsi, %rcx //将x存入%rcx
8. sarq    $63, %rcx  //将x的符号位拓展，并存入%rcx中
9. imulq   %rax, %rcx // 计算 y\_0 \* x\_63,存入%rcx中
10. imulq   %rsi, %rdx // 计算 x\_0 \* y\_63,存入%rdx中
11. addq    %rdx, %rcx // 计算 x\_0 \* y\_63 + y\_0 \* x\_63,存入%rcx中
12. mulq    %rsi       // 计算 x\_0 \* y\_0 ,结果的低位存入%rax
13. addq    %rcx,%rdx  // 计算 x\_0 \* y\_63 + y\_0 \* x\_63 + x\_0 \* y\_0,得到最终结果的高64位
14. movq    %rax,(%rdi)  //将最终结果的低64位存入%rdi的位置
15. movq    %rdx,8(%rdi) //将最终结果的高64位存入%rdi+8的位置
16. ret

**3.63**

1. **long** switch\_prob(**long** x,**long** n){
2. **long** result = x;
3. **switch**(n){
4. **case** 60:
5. result = 8 \* x;
6. **break**;
7. **case** 61:
8. **break**;
9. **case** 62:
10. result = 8 \* x;
11. **break**;
12. **case** 63:
13. result = x >>= 3;
14. **break**;
15. **case** 64:
16. result = x;
17. result <<= 4;
18. result -= x;
19. x = result;
20. **case** 65:
21. x \*= x;
22. **default** :
23. result = 75 + x;
25. }
26. **return** result;
27. }

**3.67**

A

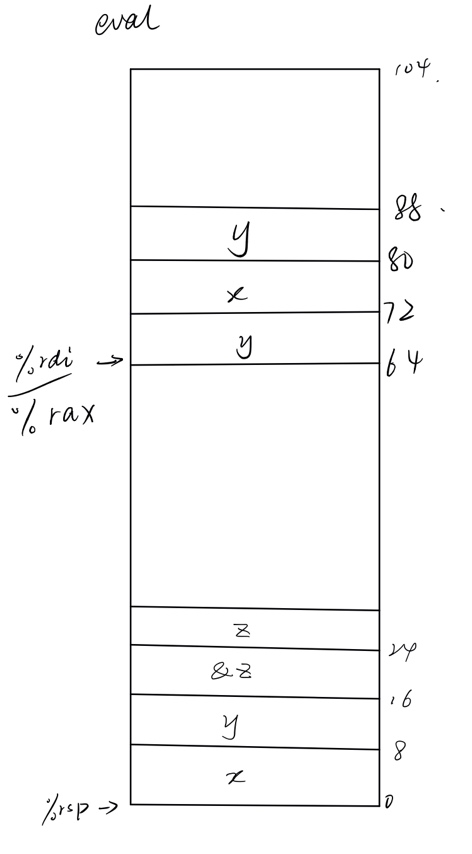


B．传递了64（%rsp）

C. 利用%rsp+偏移量的方式。

D. 从%rdi出发，按照偏移量保存

E.



F. 结构体的传递和返回，以指针形式进行，这个指针是结构内容的存储位置，也是此结构作为返回值时的返回值。

**3.71**

1. #include <stdio.h>
2. #include <stdlib.h>
3. #include <assert.h>
5. **void** good\_echo(){
6. **char** tmp[10];
7. **char** \*a;
8. **while**(1){
9. a = NULL;
10. a = fgets(tmp,10,stdin);
11. **if**(a == NULL){
12. **break**;
13. }
14. printf("%s",a);
15. //printf("\n");
16. }
17. **return**;
18. }
20. **int** main(){
21. good\_echo();
22. **return** 0;
23. }

**3.75**

A.对于第n个复数，%xmmi存放其实部，其中i = 2n-2;%xmmj存放其虚部，j = i+1。

B.%xmm0存结果实部，%xmm1存结果虚部。