Homework 4

1. Jump Table

Fill the blanks in assembly using jump table. **NOTE**: you can fill one or several instructions or symbols in one blank.

```
long x = <some value>;
long result = 0;
switch(x){
    case 5:
        result = x + 1;
        break;
    case 6: case 9:
        result = x + x;
        // fall through
    case 7:
        result = result * 5;
        break;
    case 11:
        result = x;
        break;
    default:
        result = -1;
}
return result;
```

```
    .section  .rodata
    .align 8
.L4:
    .quad ________
    .quad ________
    .quad ________
    .quad ________
    .quad ________
    .quad ________
    .quad ________
prog:
    movq  [x], -16(%rbp)
    movq  $0,  -8(%rbp)
    movq  -16(%rbp), %rax
    _______________
    jmp *_________
.L3:
    movq  -16(%rbp), %rax
    addq  $1, %rax
    _______________
.L5:
    movq  -16(%rbp), %rax
    addq  %rax, %rax
    _______________
.L6:
    movq  -8(%rbp), %rdx
    leaq  _____________, %rax
    _______________
.L7:
    movq  -16(%rbp), %rax
    _______________
.L2:
    movq  $-1, -8(%rbp)
.L8:
    movq -8(%rbp), %rax
    // function return...
```

2. Procedure call

There are two functions P and Q. ICSTA writes the assembly of these two functions. Read them and answer the following questions.

```
long Q(long n)
{
        long result;
        if (n <= 1)
                result = 1;
        else
        result = n * Q(n-1);
        return result;
}
long P(long x) {
        long a0 = x;
        long a1 = x + 1;
        long a2 = x + 2;
        long a3 = x + 3;
        long a4 = x + 4;
        long a5 = x + 5;
        long a6 = x + 6;
        long a7 = x + 7;
        h = proc(a0,a1,a2,a3,a4,a5,a6,&a7); // proc is another function
        return h;
}

Assembly of P:
P:
    pushq %r15
    pushq %r14
    pushq %r13
    pushq %r12
    pushq %rbp
    pushq %rbx
    subq $24, %rsp
    movq %rdi, %rbx
    leaq 1(%rdi), %r15
    leaq 2(%rdi), %r14
    leaq 3(%rdi), %r13
    leaq 4(%rdi), %r12
    leaq 5(%rdi), %rbp
    leaq 6(%rdi), %rax
    movq %rax, (%rsp)
    leaq 7(%rdi), %rdx
    movq %rdx, 8(%rsp)
    _______________  // you should pass the parameters to proc() here
    call proc
    ...  // then the function returns
Assembly of Q:
Q:
    movq %rdi, %r12
    movl $1, %eax
    cmpq $1, %rdi
    jle  .L35
    leaq -1(%rdi), %rdi
call Q
    imulq %r12, %rax
.L35:
    ret
```

1. Where are the local variables a0-a7 in function P stored in? Write the register name or memory address.
2. Fill the blank before *call proc* with proper instructions.
3. There is a problem in the assembly of Q. Find it out and fix it.