REX Explore

```
as -al REX64.s
```

```
      50          push   %rax
   41 50          push   %r8
      ff 30       pushq  (%rax)
   41 ff 30       pushq  (%r8)
      58          pop    %rax
   41 58          pop    %r8
      8f 00       popq   (%rax)
   41 8f 00       popq   (%r8)
      89 c0       mov    %eax,   %eax
   48 89 c0       mov    %rax,   %rax
   49 89 c0       mov    %rax,   %r8
   4c 89 c0       mov    %r8,    %rax
   4d 89 c0       mov    %r8,    %r8
      89 00       mov    %eax,   (%rax)
   41 89 00       mov    %eax,   (%r8)
   48 89 00       mov    %rax,   (%rax)
   49 89 00       mov    %rax,   (%r8)
   4c 89 00       mov    %r8,    (%rax)
   4d 89 00       mov    %r8,    (%r8)
      8b 00       mov    (%rax), %eax
   41 8b 00       mov    (%r8),  %eax
   48 8b 00       mov    (%rax), %rax
   49 8b 00       mov    (%r8),  %rax
   4c 8b 00       mov    (%rax), %r8
   4d 8b 00       mov    (%r8),  %r8
      c3          retq
```

REX Prefix

0	1	0	0	W	R	X	B

W – When 1, a 64-bit operand size is used. Otherwise, when 0, the default operand size is used

R – This 1-bit value is an extension of the ModR/M reg field

X – This 1-bit value is an extension of the SIB index field

B – This 1-bit value is an extension of the ModR/M r/m field, SIB base field, or Opcode reg field

For more information, refer to Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 2 section 2.2.1