```
.section .data
bigNr:
  .quad 0x12345678ABCDEF00

.section .text
.globl main
.type main, @function
main:
  pushq %rbp
  movq %rsp, %rbp

  # ADD
  # ===

  # add 0xFF to %al
  movq bigNr, %rax
  addb $0xFF, %al      # rax = ?

  # add 0xFF to %ax
  movq bigNr, %rax
  addw $0xFF, %ax      # rax = ?

  # add 0xFF to %eax
  movq bigNr, %rax
  addl $0xFF, %eax     # rax = ?

  # MOV and MOVZ
  # ============

  # mov  0xFF to %al
  movq bigNr, %rax
  movb $0xFF, %al
  movzbq %al, %rax     # rax = ?

  # mov 0xFF to %ax
  movq bigNr, %rax
  movw $0xFF, %ax
  movzwq %ax, %rax     # rax = ?

  # mov 0xFF to %eax
  movq bigNr, %rax
  movl $0xFF, %eax     # rax = ?
  cltq                 # rax = ?

  # MOV and MOVS
  # ============

  # mov  0xFF to %al
  movq bigNr, %rax
  movb $0xFF, %al
  movsbq %al, %rax     # rax = ?
```

```
# mov 0xFF to %ax
movq bigNr, %rax
movw $0xFF, %ax
movswq %ax, %rax      # rax = ?

# mov 0xFF to %eax
movq bigNr, %rax
movl $0xFF, %eax
movslq %eax, %rax  # = cltq         rax = ?

# mov 0xFFFFFFFF to %eax
mov bigNr, %rax
movl $0xFFFFFFFF, %eax
movslq %eax, %rax  # = cltq         rax = ?

#end
movq $0, %rax
popq %rbp
ret
```