

2.9.9 A. True. This would always yield 1 because it was asked for a two-stage comparison.

B. False. The rounding from a 32-bit two's complement representation to a double representation may not be equivalent to  $dx - dy$  if both  $x$  and  $y$  are really small and round toward a specific number.

C. True. Since the number is odd for a two's complement to a double, integer addition would be equivalent in this case.

D. False. If  $dx = \text{TMAC}$ ,  $dy = \text{TMAC} - 1$ , and  $d2 = \text{TMAC} - 2$ , this wouldn't work.

E. False. It depends on  $n$  if this is so. This case does not hold.

3.70 A.  $el.p$  -12 bytes

$el.y$  Longest Integer Return Type - long pointer -8 bytes

$e2.x$  0 bytes

$e2.next$  4 bytes

B. 16 bytes in total.

C.  $l.rdi = up$

$movl\ \%l.rdi, \%l.rax$

$leav\ (return\ addr) = \%l.rdi + 6$

$movl\ \%l.rax, \%l.rdx$

$l.rdx = (l.rax)$

$movl\ (l.rdx), \%l.rax$

$l.rdx = (l.rdx)$

$subl\ \%l.rax, \%l.rdx$

$l.rdx = l.rdx - (\%l.rax)$

val proc (min etc "up")

$up \rightarrow e2.next = * (up \rightarrow e2.next, el.p) - (up \rightarrow e2.next, el.y)$  ;

|    |   |  |
|----|---|--|
| e1 | <span style="border: 1px solid black; padding: 2px;">x p</span> | <span style="border: 1px solid black; padding: 2px;">y</span>      |
| e2 | <span style="border: 1px solid black; padding: 2px;">x</span>   | <span style="border: 1px solid black; padding: 2px;">* next</span> |