1. ANS = 0
   The IP-Stride Prefetcher, with a degree of 1, aims to predict and prefetch memory accesses based on stride patterns. However, the stride pattern in this case is alternating between +1 and +2, making it impossible for the prefetcher to accurately predict the next stride.
   As a result, the prefetcher will consistently predict the wrong stride, leading to prefetches that do not match the actual access pattern. Therefore, the coverage of the IP-Stride Prefetcher in this particular scenario is 0%, meaning it does not effectively predict and prefetch any of the memory accesses.

2. ANS = 1
   All blocks up to 'H' fit in the cache, but the arrival of block 'I' causes the Least Recently Used (LRU) algorithm to evict block 'A,' it leads to an infinite loop. Subsequent accesses to 'A' block result in block 'B' being evicted to make room for 'A' again, and this cycle continues. This use of LRU prevents the effective exploitation of spatial locality.

3. ANS = B
   One instruction is divided into five parts, (1): The opcode- As we have instruction set of size 12, an instruction opcode can be identified by 4 bits, as 2^4=16 and we cannot go any less. (2) & (3): Two source register identifiers- As there are total 64 registers, they can be identified by 6 bits. As they are two i.e. 6 bit + 6 bit. (4): One destination register identifier- Again it will be 6 bits. (5): A twelve bit immediate value- 12 bit. Adding them all we get, = 4 + 6 + 6 + 6 + 12 = 34 bit = 34/8 byte = 4.25 byte As given Each instruction must be stored in memory in a byte-aligned fashion,4.25 is not byte alignment, memory address should be 0,1,2,3,4,5,6,7..... so it should be 5 bytes. As there are 100 instructions, we have a size of 5*100= 500 bytes. Hence (B) 500 is the answer.