```solidity
1  contract EtherAuction {
2    //Anyone can bid by calling this function and supplying the
   ↪ corresponding eth
3    function bid() public payable {
4      require(auctionStarted);
5      require(now < auctionEndTime);
6      require(msg.sender != auctioneer);
7      // If sender is already the highest bidder, reject it.
8      require(highestBidder != msg.sender);
9      address _newBidder = msg.sender;
10     uint previousBid = balances[_newBidder];
11     uint _newBid = msg.value + previousBid;
12     // Each bid has to be 0.05 eth higher
13     if (_newBid  == highestBid + (5 * 10 ** 16)) return;
14     // The highest bidder is now the second highest bidder
15     secondHighestBid = highestBid;
16     secondHighestBidder = highestBidder;
17     highestBid = _newBid;
18     highestBidder = _newBidder;
19     latestBidTime = now;
20     // Update the bidder's balance so they can later withdraw
   ↪ any pending balance
21     balances[_newBidder] = _newBid;
22   }
23 }
```