Display start menu;  
IF option 1 (Load Bids) is chosen {  
 OPEN .csv file;  
 READ each line and convert relative fields to bid info;  
 APPEND *bids* vector with each bid; }  
  
IF option 2 (Display All Bids) is chosen {  
 FOR each bid in *bids* vector:  
 PRINT the bid’s info; }  
  
IF option 3 (Selection Sort All Bids) is chosen {  
 FOR each bid in *bids* vector starting with i = 0:  
 SET *smallestBidIndex* variable equal to *i;* FOR each bid in *bids* vector starting with j = i +1:  
 IF the bid @ index *j* is less than the bid @ index *i:*  
 SET *smallestBidIndex* equal to j;  
 ELSE:  
 continue;  
 SET *tempBid* variable equal to bid @ index *i;* SET bid @ index *i* equal to bid @ index *smallestBidindex;* SET bid @ index *smallestBidIndex* equal to *tempBid;* }  
  
IF option 4 (Quick Sort All Bids) is chosen (*bids* vector, *begin* index, *end* index) {  
 SET variable *j* equal to 0;  
 IF *begin* index is greater than or equal to *end* index:  
 return out of the function; (NOTE: this is your base case/when function has completed);  
 SET variable *j* to call the *Partition* function, passing (*bids* vector, *begin* index, *end* index);  
  
 PARTITION(*bids* vector, *begin* index, *end* index): {  
 SET *pivotBid* equal to the bid @ the index between *begin* index and *end* index;  
 INITIALIZE *low, high,* and *done* helper variables;  
 SET *low* equal to *begin, high* equal to *end*, and *done* equal to *false;* WHILE *done* is *false:* {  
 WHILE bid @ index *low* < *pivotBid:* {increment *low*};  
 WHILE bid @ index *high* > pivotBid: {decrement *high*};  
 IF index *low* >= index *high:   
 done =* true(EXITS the partition function);  
 ELSE:  
 INITIALIZE *tempHolder* bid and set equal to the bid @ index *low;* SET the bid @ index *low* equal to the bid @ index *high;* SET the bid @ index *high* equal to the bid *tempHolder;* INCREMENT *low* and DECREMENT *high; }* return *high;* }  
  
 CALL quicksort(*bids* vector, *begin* index, *j* index) recursively;  
 CALL quicksort(*bids* vector, *j+1* index, *end* index) recursively; }  
  
 IF option 9 (Exit) is chosen: {PRINT exit message and EXIT}