

Loop Body: Find the Place to Insert the New Card

New card is at position (`sorted - 1`).

```
current = values[sorted - 1];
// find place to insert new card
values[index] = current;
```

After inner loop, `index` is correct position for new card.

Loop Over Possible Positions Until Correct One is Found

Look first at new card's original position.

```
for (index = sorted - 1; 0 < index;
    index--) {
    // check possible position
    values[index] =
        values[index - 1];
```

Check one position per iteration.

Move card over if new card is smaller.

Position Check Simply Compares Card Values

```
if (current >= values[index - 1]) {
    break;
}
```

If the new card's value is at least as great as the card in the previous position, stop searching.

Reference Copy of Insertion Sort

```
void insertion_sort (int32_t values[], int32_t num_vals)
{
    int32_t sorted, current, index;
    for (sorted = 2; num_vals >= sorted; sorted++) {
        current = values[sorted - 1];
        for (index = sorted - 1; 0 < index; index--) {
            if (current >= values[index - 1]) {
                break;
            }
            values[index] = values[index - 1];
        }
        values[index] = current;
    }
}
```