Bubble Sort

```
void sortArray(int list[], int size)
{
	bool swap;
	int temp;
	do {
		swap = false;
		for (int count=0; count < (size-1); ++count) {
			if (list[count] > list[count+1]) {
				temp = list[count];
				list[count] = list[count+1];
				list[count+1] = temp;
				swap = true;
			}
		}
	} while (swap);
}
```

Could use a function to swap values:

```
void swapValues(int& a, int& b)
{
	int temp=a;
	a = b;
	b = temp;
}
void sortArray(int list[], int size)
{
	bool swap;
	do {
		swap = false;
		for (int count=0; count < (size-1); ++count) {
			if (list[count] > list[count+1]) {
				swapValues(list[count], list[count+1]);
				swap = true;
			}
		}
	} while (swap);
}
```

Could be more efficient to avoid checking every value in the array:

```
void sortArray(int list[], int size)
{
	bool swap;
	int total=size;
	do {
		swap = false;
		for (int count=0; count < (total-1); ++count) {
			if (list[count] > list[count+1]) {
				swapValues(list[count], list[count+1]);
				swap = true;
			}
		}
		total--;
	} while (swap);
}
```

Selection Sort

```
void selectionSort(int list[], int size)
{
        int minIndex;
        int minValue;
        for (int startScan=0; starScan < (size-1); ++startScan) {
                minIndex = startScan;
                minValue = list[minIndex];
                for (int index = startScan+1; index < size; ++index) {
                        if (list[index] < minValue) {
                                minValue = list[index];
                                minIndex = index;
                        }
                }
                list[minIndex] = list[startScan];
                ist[startScan] = minValue;
        }
}
```