range class

```
class range {
private:
  int _start;
  int _stop;
  int _stride;

public:

  // supports trivial range(3) to be shorthand for {3}
  range(int index)
    : _start(index), _stop(index+1), _stride(1) { }

  // supports construction such as range(3,6), which includes values {3, 4, 5}
  range(int start, int stop)
    : _start(start), _stop(stop), _stride(1) { }

  // supports construction such as range(3,2,8), which includes values {3, 5, 7}
  range(int start, int stride, int stop)
    : _start(start), _stop(stop), _stride(stride) {

    if (stride < 1)
      throw invalid_argument("stride must be positive.");
  }

  // Returns starting index
  int start() const {
    return _start;
  }

  // Returns stopping index
  int stop() const {
    return _stop;
  }

  // Returns stopping index
  int stride() const {
    return _stride;
  }

  // Returns the number of values included within the range
  int size() const {
    // partials strides should count as one.  e.g. range(1,2,4).size() should be 2
    if (_stop >= _start)
      return (_stop - _start + _stride - 1) / _stride;  // trucates properly
    else
      return 0;   // e.g., range(5,3)
  }

};
```