File Date.h: **You may remove and keep this page of the exam.**

```
class Date {

private:

  long dayNumber;  // 1 = Jan 1, 1900 and so on

public:

  // This class ensures that dates are always valid and always lie between
  // Jan 1, 1900 and Dec 31, 2099.
  // Attempts to move a date out of this range produce the appropriate
  // limiting value.  Invalid constructor arguments, read errors, and so on
  // result in Jan 1, 1900.

  // constructs a date containing Jan 1, 1900
  Date ();

  // constructs a date containing the specified date.
  Date (int day, int month, int year);

  // reads a date (dd mm yyyy format) from the given input stream. if an
  // error of any sort (bad read, invalid date) occurs, the input stream is
  // left in the failed state.
  void read (istream &is);

  // writes a date to the specified output stream (in dd/mm/yyyy format)
  void write (ostream &os) const;

  // moves a date the specified number of days forward (positive day
  // values) or backwards (negative day values).
  void move (int days);

  // returns the day of the week (1 = Monday, etc.)
  int dayInWeek () const;

  // returns a date in dd, mm, yyyy form
  void getDDMMYYYY (int &day, int &month, int &year) const;

  // returns the difference between two dates.  this will be positive if
  // the other date is earlier than the implicit date (and negative if the
  // other date is later)
  int daysAfter (const Date &otherDate) const;

  // returns 1 if the date is greater than (later than) the other date.
  // returns -1 if the date is less than (earlier than) the other date.
  // returns 0 if the two dates are equal
  int compareTo (const Date &otherDate) const;

};
```