```
#ifndef STRUCTFILE_H_INCLUDED
#define STRUCTFILE_H_INCLUDED

///Declare and define the date, author, and book structs and their respective attributes
struct date
{
   int month;
   int day;
   int year;

   date() ///default constructor
   {
      month = 0;
      day = 0;
      year = 0;
   }
};

struct author
{
   std::string name;
   date d; ///The author struct will have a date struct within it
   std::string mostNoteBook;
   std::string nationality;

   author() ///default constructor
   {
      name = " ";
      date d();
```

```
    mostNoteBook = " ";
    nationality = " ";
  }

  bool dateOlderThan(int otherDate); ///member function declarations
  void printAuthorsToFile(author* a, date* d, int &n);
};

struct book
{
  std::string bookTitle;
  std::string authorName;
  int englishPubYear;
  std::string originalLang;
  std::string genre;
  int totalSales;

  book() ///default constructor
  {
    bookTitle = " ";
    authorName = " ";
    englishPubYear = 0;
    originalLang = " ";
    genre = " ";
    totalSales = 0;
  }

  void sortBookSales(book* b, int &n); ///member function declarations
  bool salesGreaterThan(int otherSalesNum);
```

```
   void printBooksToFile(book* b, int &n);
};

struct library
{
   author a[100];
   book b[100];

   library() ///default constructor
   {
      a[100];
      b[100];
   }

   void initializeLibrary(library* l, struct book* b1, struct author* a1, struct date* d1, int &n, int &m);
///member function declarations
};

void sortAuthorDate(struct author* a, struct date* d, int &n);

void initializeBookArray(struct book* b, int &n); ///Declare the required functions

void printBookArray(struct book* b, int &n);

void initializeAuthorArray(struct author* a, struct date* d, int &n);

void printAuthorArray(struct author* a, struct date* d, int &n);

void matchAuthors(struct book* b, struct author* a);
```

```
#endif // STRUCTFILE_H_INCLUDED
```