

San José State University
Department of Computer Engineering

CMPE 152 Compiler Design

Sections 4 and 5
Spring 2018
Instructor: Ron Mak

Assignment #1

Assigned: Thursday, January 25
Due: Friday, February 2 at 11:59 pm
Individual assignment, 100 points max

Write a Pascal program

The purpose of this assignment is to give you a crash course in Pascal programming.

Input file

The Library Books application reads a text file as input and generates a report from the input data. Here is the input file, which has two sections:

```
5
123456 Ancient Programming Language Pascal
246800 How to Manage Big Hairy Legacy Applications
654321 Modern Programming Language C++
707070 Which is Better, C++ or Java?
999999 Compiler Writing for Fun and Profit
8
123456 John Smith
999999 Mary Jane
246800 Sarah Monroe
654321 Frank Jones
999999 Arnold Brown
654321 Kathy Wilson
123456 John McDonald
999999 Leslie Nelson
```

The first section consists of library book records, one per line. Each line has a 6-digit book ID, range 100000-999999, and after one or more blanks, the book title. Preceding this section is the number of book records.

The second section consists of borrowing activity records, one per line. Each line has the ID of the book being borrowed, and after one or more blanks, the name of the borrower. Preceding this section is the number of activity records.

You may assume the input file does not contain any errors: no invalid numbers and the ID in each activity record is valid.

Input file: <http://www.cs.sjsu.edu/~mak/CMPE152/assignments/1/LibraryBooks.in>

Expected output

Given this input file, your program should generate the following output:

```
Library book application

Number of books = 5
 123456: Ancient Programming Language Pascal
 246800: How to Manage Big Hairy Legacy Applications
 654321: Modern Programming Language C++
 707070: Which is Better, C++ or Java?
 999999: Compiler Writing for Fun and Profit

Borrowing activity
 123456 borrowed by John Smith
      Ancient Programming Language Pascal
 999999 borrowed by Mary Jane
      Compiler Writing for Fun and Profit
 246800 borrowed by Sarah Monroe
      How to Manage Big Hairy Legacy Applications
 654321 borrowed by Frank Jones
      Modern Programming Language C++
 999999 borrowed by Arnold Brown
      Compiler Writing for Fun and Profit
 654321 borrowed by Kathy Wilson
      Modern Programming Language C++
 123456 borrowed by John McDonald
      Ancient Programming Language Pascal
 999999 borrowed by Leslie Nelson
      Compiler Writing for Fun and Profit

Activity summary
 123456 borrowed 2 time(s): Ancient Programming Language Pascal
 246800 borrowed 1 time(s): How to Manage Big Hairy Legacy Applications
 654321 borrowed 2 time(s): Modern Programming Language C++
 707070 borrowed 0 time(s): Which is Better, C++ or Java?
 999999 borrowed 3 time(s): Compiler Writing for Fun and Profit
```

Pascal program requirements

Your Pascal program must have the following features:

- A record data type to contain the information for each book.
- Procedures and/or functions.
- At least one parameter passed by reference.
- **Do not use the built-in string type.** Represent a string as an array of characters, as in classic Pascal. For example:

```
TYPE
    Field = ARRAY [1..12] OF char;
```

A string value can only be read one character at a time from the input and entered into the character array. However, an entire string (the entire array) can be written all at once. Blank-fill the end of a string to the length of the array.

Text input hints

- If `ch` is a character variable, then `read(ch)` will read the next character from the standard input.
- A call to `readln` will skip the rest of the current input line. Then the next call to `read` will read starting on the next input line.
- Boolean functions `eofln` and `eof` without parameters test whether the standard input is at the end of the current line or at the end of file, respectively. (If you're typing input data into a Windows command window, a line containing only control-Z is the end-of-file marker. On Linux and the Mac, it's control-D.)

Your program, including comments and blank lines, should have fewer than 200 lines. Even though Pascal is a simple language compared to Java, learning a new language can be tricky. **Do not wait until the last minute** to work on this assignment!

What to submit to Canvas

- Your reasonably commented Pascal source file **LibraryBooks.pas**, as a text file.
- A text file of your program's output report.

Zip the two files together and submit the zip file into Canvas: **Assignment #1**

This is an individual assignment. Your program must be your own work.

Rubric

Your program will be graded according to these criteria:

Criteria	Maximum points
Good output <ul style="list-style-type: none">• Correct output values.• Correct report format.	40 <ul style="list-style-type: none">• 20• 20
Good program design <ul style="list-style-type: none">• Record data type.• Procedures and/or functions.• At least one parameter passed by reference.	30 <ul style="list-style-type: none">• 10• 10• 10
Good program style <ul style="list-style-type: none">• Descriptive variable names and meaningful comments.• Good use of control statements.• In general, well written.	30 <ul style="list-style-type: none">• 10• 10• 10

A quick tour of the Lazarus Pascal IDE

A very good IDE (integrated development environment) you can use to develop Pascal program is Lazarus, which you can download for Linux, Mac, or Windows:

<https://www.lazarus-ide.org/index.php>

Download and install Lazarus. After you start it for the first time, close all of its windows except for the main menu bar. Select the menu item File | New. In the popup window, select Program (under Project) and click the OK button. An editor window with the name project1.lpr will open. Delete all the default text. Select the menu item File | Save As and choose the name **LibraryBooks.pas**.

Now you can edit your program and save it periodically. To compile, select the menu item Run | Build. If there are no syntax errors, Lazarus will generate an executable file (Widgets.exe on Windows) which you can run on the command line as shown above.

You can also run your program from within Lazarus. Select the menu item Run | Run. A console window will open into which you can type (or paste) the input data. There are advantages of running within Lazarus: you can set breakpoints, look at the values of local variables, get the runtime stack, etc. Unfortunately, the console window (which contains the program output) disappears immediately when your program finishes.

If Lazarus doesn't work well for you, you can always use the command-line Free Pascal compiler `fpc`. For example,

```
fpc LibraryBooks.pas
```

A very useful link: Pascal runtime error codes: <https://www.freepascal.org/docs-html/user/userap4.html>