Parsing A Flat File, Reading

In computer programming, "parsing" means to process a string and pull data items out from it. Our schedule history TXT file has this line, for example:

```
Spring 2016\t8301\tCOMSC-210\tBurns\tW 7:00-9:50pm L-142
```

The **\t**'s are tab characters. What we want out of this line are 6 things:

1. the term (Spring 2016) as a C++ string,
2. the section number (8301) as a C++ string,
3. the course (COMSC-210) as a C++ string,
4. the instructor (Burns) as a C++ string,
5. the day/time/room (W 7:00-9:50pm L-142) as a C++ string, and
6. the subject code (COMSC) as a C++ string.

The first 5 are done by searching for the tab characters and copying the text between tabs. The last is done by finding the dash in the course, and copying out the text in front of the dash.

Here is a CPP that reads and parses the TXT file:

```
#define _CRT_SECURE_NO_WARNINGS

#include <fstream>
#include <iostream>
#include <string>
using namespace std;

#include <cstring> // for strtok and strcpy

int main()
{
  // for parsing the inputfile
  char* token;
  char buf[1000];
  const char* const tab = "\t";

  // open the input file
  ifstream fin;
  fin.open("dvc-schedule.txt");
  if (!fin.good()) throw "I/O error";

  // read the input file
  while (fin.good())
  {
    // read the line
    string line;
    getline(fin, line);
    strcpy(buf, line.c_str());
    if (buf[0] == 0) continue; // skip blank lines

    // parse the line
    const string term(token = strtok(buf, tab));
    const string section(token = strtok(0, tab));
    const string course((token = strtok(0, tab)) ? token : "");
    const string instructor((token = strtok(0, tab)) ? token : "");
    const string whenWhere((token = strtok(0, tab)) ? token : "");
    if (course.find('-') == string::npos) continue; // invalid line: no dash in course name
    const string subjectCode(course.begin(), course.begin() + course.find('-'));

    // if I get this far, then it's a valid record
    cout << term << '|' << section << '|'
      << course << '|' << instructor << '|'
      << whenWhere << '|' << subjectCode << endl;
  }
  fin.close();
}
```

The **#define _CRT_SECURE_NO_WARNINGS** is to suppress compiler warnings on Microsoft compilers about using the "unsafe" **strtok** C library function. Don't worry -- it's safe.

Where To Put Input Files

When we use files for *input* in this course, the files are expected to be in the "working" folder -- that is, your program should be able to open the file *without* a "path" designation. For command line compiling, it's easy -- just put the input file where the CPP is located. But for XCode and Code::Blocks it's not so easy.

In **XCode**, here's where to find the folder for input (and output) files for a project: In Finder, use Go -> Go to folder... and type **~/Library**. Then drill down through the folders like this:

1. Developer
2. XCode
3. DerivedData
4. (the name of your project + an ID tag)
5. Build
6. Products
7. Debug (or Release) -- *that's where input and output files go!*

To see how it's done in **Code::Blocks**, watch this video: