CSCI 201: Introduction to Algorithm Design

Problem #7 -- Due 1 May, 1991

[35 points]

In order to speed up the processing of mail, the United States Post Office has requested that we try to use nine digit ZIP codes. These nine digit ZIP codes are written as five and four digit numbers separated by a hyphen, as in 28804-3299. These nine digit ZIP codes are then transformed into a machine readable optical bar code. You may have noticed these bar codes on the lower right corner of mail you have received. They consist of a series of long and short bars. For example, the optical bar code for 28804-3299 is

```
|..|.||..|.|..|.||....|..|..|.|..|.||.|..|.|...|.|.|
```

Note that there are 52 vertical bars in the above code. The beginning and ending vertical bars are always long. The middle 50 bars are used to encode 10 digits. (I'll explain why 10 rather than 9 in a minute). Five vertical bars are used for each digit. The code used for each digit is given in the following chart:

```
1 = ...||        6 = .||..
2 = ..|.|        7 = |...|
3 = ..||.        8 = |..|.
4 = .|..|        9 = |.|..
5 = .|.|.        0 = ||...
```

[You might ponder this for a pattern.] This particular encoding is called a two-of-five code by computer scientists because each digit is encoded with five bars, two of which are long. Thus the optical bar code for 28804-3299 can be decrypted as:

```
| ..|.| |..|. |..|. ||... .|..| ..|.| ..|.| |.|.. |.|.. .|.|. |
   2     8     8     0     4     3    2     9     9     5
```

Now, we can explain the 10th digit, in this case a 5. The tenth digit is a "check digit". It is chosen so that if you add up all ten digits, the resulting number is divisible by ten. For example, 2+8+8+0+4+3+2+9+9+5 equals 50, which is divisible by ten. The machines that sort mail always add up the ten digits of the optical code to make sure the sum is divisible by ten. If it isn't, the machines assume that ZIP code was either improperly encoded or improperly read.

Finally, let's get to your assignment. You're actually going to write two programs. The first program will read input lines of the following form:

Asheville, NC 28804-3299

It will *parse* the input line and store it into a Pascal record with at least three fields: city, state (a two letter abbreviation), and ZIP code (nine digit). Your program will then write this record to a file. If your program reads an improper input line, it should print an error message and, of course, not add the record to the file. Your program should read records until it reaches the end of input file, *e.g.*, **EOF(input)**. By the way, you end terminal input by typing **^Z**.

Your second program will read input from the terminal and the file generated by your first program. The input from the terminal will be a series of line containing optical codes, *e.g.*,

|..|.||..|.|..|.||....|..|..|.|..|.||.|..|.|...|.|.|

Your program will check these optical codes to see if they are of the proper format. For example, it will make sure they are the right length and have the appropriate checksum. Your program will then transform the optical code into a nine-digit ZIP code and search the record file generated by your first program to find the city and state for that particular ZIP code. Again, have your program read until it reaches the end-of-file of **input**.

Hints and Suggestions

The following function **CheckDigit** computes the tenth check digit in the optical encoding of ZIP codes used by the USPO. It's input parameter is a single integer between 0 and 999,999,999. **CheckDigit** returns the appropriate check digit for that ZIP code. (And I think it ought to work.)

```
FUNCTION CheckDigit(I: INTEGER): INTEGER;

BEGIN
  VAR
    Irem: INTEGER ;
    Isum: INTEGER ;

    Irem := I;
    Isum := 0;

    WHILE (Irem > 0) DO
      BEGIN
        Isum := Isum + Irem MOD 10 ;
        Irem := Irem DIV 10
      END;

     CheckDigit := 10 - Isum MOD 10
  END ;
```

There will be a temptation to restrict the number of address lines your programs process. Try to avoid that temptation by the use of linked lists. (No, it's not required, but it can't hurt.)

Be on the lookout for envelopes with optical codes to use in testing your programs.