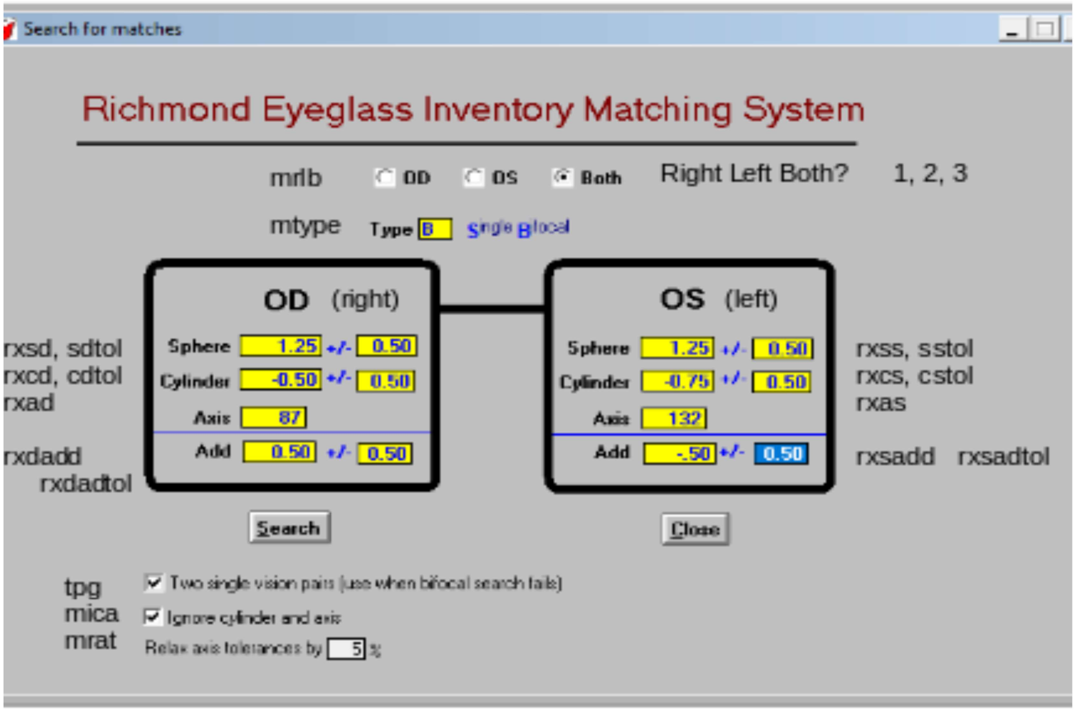
The PhilScore: Translating the original Foxpro for Windows code into a readable algorithm

REIMS (Richmond Eyeglass Inventory Matching System) was invented and implemented by the late Dr. Philip Richmond, OD, and the late Skip Charles, the programmer. It stores an inventory of donated glasses, along with their characteristics. When a new eyeglass prescription is issued, it can be entered into the REIMS search screen (see below) and the closest matching eyeglasses are displayed in rank order by "PhilScore." If a match cannot be made on the first search, the tolerances can be widened, or two single pairs can be located for a bifocal prescription.



Search for matches

Richmond Eyeglass Inventory Matching System

○ OD ○ OS ● Both

Type B Single Bifocal

OD (right)				OS (left)			
Sphere	1.25	+/-	0.50	Sphere	1.25	+/-	0.50
Cylinder	-0.50	+/-	0.50	Cylinder	-0.75	+/-	0.50
Axis	87			Axis	132		
Add	0.50	+/-	0.50	Add	-.50	+/-	0.50

☑ Two single vision pairs (use when bifocal search fails)

☑ Ignore cylinder and axis

Relax axis tolerances by 5 %

Figure 1: REIMS Inventory Search Screen, annotated with FoxPro variable names

The sample main inventory table I was provided with is GLSKU, which contains 4991 records, last updated in 2014. The columns of data in it are:

Field	Field Name	Type	Width	Dec	Index	Collate	Nulls
1	SKU	Character	20				No
2	TYPE	Character	1				No
3	ODSPHERE	Numeric	6	2			No
4	ODCYLINDER	Numeric	6	2			No
5	ODAXIS	Numeric	3				No
6	ODADD	Numeric	6	2			No
7	OSSPHERE	Numeric	6	2			No
8	OSCYLINDER	Numeric	6	2			No
9	OSAXIS	Numeric	3				No
10	OSADD	Numeric	6	2			No
11	GENDER	Character	1				No
12	MATERIAL	Character	1				No
13	SIZE	Character	1				No
14	TINT	Character	1				No
15	ENTERDATE	Date	8				No

SKU is the stock-keeping number
TYPE is (B)ifocal or (S)ingle
The OD/OD measurements are the lens characteristics (d=dextral/right, S=sinestral/left)
Gender is (M)ale, (F)emale or (U)nisex
MATERIAL is blank, (M)etal or (P)lastic
SIZE is (L)arge, (M)edium, (S)mall or (C)hild
TINT is blank, (N)one, (L) or (D)
ENTERDATE is the date the record was entered.

To transfer these records into a different database system, I recommend opening (a copy!) of the data file in Excel, and using the Excel Export functions to export a comma-separated-value (CSV) formatted file, which more database systems can readily read.

The search processing

When the “Search” button is pressed, the search process begins. First, it sorts out which type of prescription to find, right, left or both. The process is the same in each case, only for the specified lenses.

If the checkbox for “Ignore cylinder and axis” is checked, the inventory is narrowed to lenses that match the sphere +/- the tolerance, and the cylinder and axis are ignored, go to step 4 below, where the results are ranked and displayed.

If that checkbox is NOT checked, the lens inventoried is narrowed to those that:
1) match the sphere +/- tolerances as above
2) match the cylinder +/- tolerances
2) Pass the AtoLTF() test, explained below,
OR
3. match one of 3 widened tolerances as determined by the SPEQ() function, explained below.
4. Now that all possible glasses are selected, they are ranked in order using the RANK() function, see below, which returns the PhilScore.

The AtoLTF Test

AtoLTF test (zc, za, zac) for (lens cylinder, rx axis, lens axis) where I refer to ‘lens’ as the actual measured glasses lens, and rx refers to the prescription entered on the search screen. There are several steps:

Step 1: Given the cylinder of the glasses’ lens (zc), calculate a ‘zat’ value:

Lens cylinder less than or equal to:	zat
-4	7
-3	8
-2.5	8
-2	9
-1.75	10
-1.50	10
-1.25	13
-1.00	15
-0.75	20
-0.50	25
-0.25	35
Otherwise	90

round the zat number to 100% plus “relax axis tolerances by” figure (mrat on the screen). If the original zat was 90, leave it at 90.

Step 2: If the zat is less than 90, increase zat by the “relax axis tolerances” percent. So, if lens cylinder was -1.75 and the tolerance was 10%, the zat of 10 would be increased by 10% to 11.

Step 3: using the zat from Step 2 and the prescription axis za), run three tests:

Test	Zac (lens axis) between this:	and this:	OR	Zac (lens axis) between this:	and this:
If za+zat > 180	0	za+zat-180		Za-zat	180
If Za-zat < 0	0	Za+zat	OR	180 – (zat-za)	180
Otherwise,	Za-zat	za+zat	OR	999	999

If the lens and prescription pass ANY of the three tests, the glasses can be considered, otherwise they are rejected.

SPEQ Function

FUNCTION SPEQ, from file SPEQ.PRG:

The SPEQ function is passed in the prescription Sphere, Cylinder and Axis values, and returns a set of one, two or three pairs of upper and lower bounds for cylinder and sphere values, depending on the cylinder value. Axis values seem to be ignored

Rx Cylinder	C1 = rxc	S1 = rxs	c2 = rxc	s2 = rxs	c3 = rxc	s3 = rxs
<= -1.5	+0.5	-0.25	+1	-0.5	+1.5	-0.75
<= -1.0	+0.5	-0.25	+1	-0.5	10*	10*
<= -0.5	+0.5	-0.25	10*	10*	10*	10*

* I’m guessing that throwing a value of ten in there would disqualify the lens for consideration.

* original source code:

```
PROCEDURE speq
 PARAMETER rxs, rxc, rxa
 STORE 10 TO s1, s2, s3, c1, c2, c3
 DO CASE
    CASE rxc<=-1.5
       c1 = rxc+0.5
       s1 = rxs-0.25
       c2 = rxc+1
       s2 = rxs-0.5
       c3 = rxc+1.5
       s3 = rxs-0.75
    CASE rxc<=-1
       c1 = rxc+0.5
       s1 = rxs-0.25
       c2 = rxc+1
       s2 = rxs-0.5
    CASE rxc<=-0.5
       c1 = rxc+0.5
       s1 = rxs-0.25
 ENDCASE
```

PhilScore: all the glasses that have passed the previous tests are assembled in in a list (a CURSoR or “CURrent Set of Records” in FoxPro terms) and then are sorted by Philscore through the RANK() function:

```
rank(mtype,'OD',rxsd,rxcd,rxdadd,0,0,0,odsphere,odcylinder,odadd,0,0,0) AS philscore
```

mtype: ‘B’Bifocal or single prescription lens
OD/OS/OU: for dextral (right lens), sinestral (left) or both lenses
rxsd: prescription sphere right
rxcd: prescription cylinder right
rxdadd: added right
rxss, rxcs, rxsadd … same for left, substitute ‘s’ for the last ‘d’ then:
odsphere: measured lens sphere
odcylinder: measured lens cylindeer
odadd: measured add
… same for left, substitute ‘s’ for the first ‘d’

The ranking algorithm returns an ‘ind’ index value that’s used to sort the results of all of the glasses found to match the prescription, type, right/left/both eyes and tolerances.

Here’s the math for the right eye (OD) version:

Initial index is the sum of:
1. unsigned difference between lens and prescription sphere
2. unsigned difference between lens and prescription cylinder
3. If the prescription is a bifocal, one-tenth the unsigned difference between lens and prescription added strength

and one of these two:
4a. if the unsigned difference between lens and prescription is over 90, 180 minus that difference, divided by 3600.
4b. if the unsigned difference between lens and prescription less than or equal to 90, that difference divided by 3600.

5. That initial index is modified if any of the following are true:

If the lens sphere is greater than zero:	
If (a) the prescription sphere minus the lens sphere is equal to one half of (the lens cylinder minus the prescription cylinder) and (b) the prescription sphere is greater than the lens sphere and (c) the unsigned difference of the lens cylinder and the prescription cylinder is less than one	subtract 0.55 from the index
If (a) the prescription sphere equals the lens sphere and (b) the unsigned difference of the prescription and lens cylider is 0.75 or less and (c) lens cylinder and the prescription cylinder are not equal	subtract 0.12 from the index
If (a) the lens sphere is greater than the prescription sphere and (b) the prescription cylinder is greater than the lens cylinder OR (a) the lens sphere is less than the prescription sphere and (b)the prescription cylinder is less than the lens cylinder	If the unsigned difference of the sphere is the same as that of the cylinder: (a) if the absolute difference between lens and prescription sphere is identical to lens and prescription cylinder, then: (i)subtract 0.30 if the unsigned difference in cylinder lens and prescription is less than 0.5 (ii) subtract (11/20) if the unsigned difference in cylinder lens and prescription is more than 0.5 (b) if NOT (a) above: (I) subtract 0.25 if (a)(i) applies (ii) subtract 0.5 if (a)(ii) applies
If the lens sphere is not greater than zero:	ALL OF THE ABOVE Step 5 EXCEPT the first case is a smaller change to the index:
If (a) the prescription sphere minus the lens sphere is equal to one half of (the lens cylinder minus the prescription cylinder) and (b) the prescription sphere is greater than the lens sphere and (c) the unsigned difference of the lens cylinder and the prescription cylinder is less than one	subtract 0.50 from the index

6. If the prescription is for bifocal and the lens axis is greater than the prescription axis, subtract from the index one-one-hundredth of the difference of the lens axis minus the prescription axis.

7. if the prescription sphere is greater than the lens sphere and the prescription sphere is greater than zero, ADD 0.25 to the index.

At this point, the RANK() function returns the index, the Philscore for a right-eye-only prescription.

For a left-eye only, type ‘OS’ the exact same math is used.

For a two eye prescription, the index is calculated using the same 7 steps. In each step, the calculation is made for the right (“OD”) side and then the left (“OS”) and added together.