Fair and square

The complicated task of allocating work shift and holiday rosters that are fair to all can be eased by using a spreadsheet. Stephen Wells shows how.

Andy Christou works for one of the largest companies in the country. At least, it was until it was denationalised and thousands of people were made redundant. But that's an interpolated comment of mine.

Andy's problem is as follows: "*We used to work a normal five-day, 40-hour week. To improve service to our customers it was decided to extend the working day but at no cost. The method chosen was to only work for nine out of the ten business days in a fortnight so that at the end of two weeks each person had still put in only 80 hours. This was christened the 'nine-day fortnight'.*

"*In order that Mondays and Fridays were shared fairly, we had a 'rolling day off' pattern: if you had a Monday off, your next day off would be two weeks and a day away, i.e. Tuesday. This created another problem because after nine periods of having a day off, you ended up with the Friday of one fortnight being next to the Monday of the next fortnight which was not acceptable to our managers.*

"*The final method was as follows: if the last day off was a Monday, then the next day off would be Tuesday fortnight,*

i.e. 15 days away; if the day off was Tuesday, the next day off would be Wednesday fortnight, i.e. 15 days away; if the day off was Wednesday, the next day off would be Thursday week, i.e. eight days

away; if the day off was Thursday, the next day off would be Friday fortnight, i.e. 15 days away; and if the day off was Friday, the next would be Monday fortnight, that is 17 days away.

"*My problem is, given a starting point of 1st January 1996, how do I implement the above so that a worksheet calculates the next day off due and then marks the relevant cell with 'SDO'?*

"*I realise that I have to carry out a test to see what the day of the week is and then*

move the cursor along as appropriate and mark the cell. All the initial days would be entered manually."

I asked Andy to lay out a dummy worksheet to illustrate what he hopes to

Fig 1 The first two weeks (starting 30/12/95) of part of a schedule for employees' special days off

achieve. The starting corner of Andy's worksheet is shown in Fig. 1.

In the Navy

Another staff-scheduling problem is posed by Malcolm Campbell: "*I work at a large Naval establishment where we use lieutenants, sub-lieutenants and warrant officers as duty personnel. I organise a work roster in three month blocks, for out-of-hours managerial duty cover for every day of the week (including weekday*

Wrong number

You may recall from an earlier column that a reader emailed me to inquire whether I was the same Stephen Wells who wrote for *New Musical Express*. Now I've received a wedding invitation from someone who's seen my name here. Not that I know the couple… nor do I have a partner called Quiana. What I'm hoping for is third time lucky: it would be very nice if a solicitor were looking for a Stephen Wells who is a major legatee.

evenings, weekends and national holidays) for 30-35 managers.

"We don't get extra pay or time off for these duties and we have to sleep on site, so you can imagine how unpopular this duty is! I come up against the human factor: people are away travelling; some people wish to do extra WDs (work days) to get WEs (weekends) off; others can't do particular days due to outside commitments; some are away travelling so much in previous periods that they are granted a reduced number of duties this period, and so on.

"In an effort to be fair to all, I try to spread the load, i.e. give everyone the same number of WDs and WEs. This is fantastically difficult and I revert to graph paper, eraser and pencil. It would seem logical to assign a weighting to each type of day, say two for a WD, three for a Friday and four for a WE or national holiday. Ideally, the spreadsheet would work out an average score and then try many permutations to get everyone's score as close as possible to each other (for the ones on full duties only)."

I asked Malcolm to supply a dummy worksheet and he provided Fig 2. Both

Fig 2 The start of a holiday schedule for Navy managers. It's based on many ifs, ands, and buts

EXCELlibrary

I'm currently looking for a source for review copies of new books on spreadsheets and related subjects. Until then I will adopt the strategy of Neil Negandhi who maintains The Unofficial Microsoft Excel Page. (It's called unofficial but you reach it via a link from Microsoft's Excel web page.) Neil lists current books which have been recommended by his readers, viewers, hits or whatever a web page receiver is called.

■ At the top of his list is *Excel for Windows 95 Bible* by John Walkenbach, published by IDG Books Worldwide, ISBN: 1-56884-495-6. It has 990 pages and includes a CD-ROM ($39.95 in the US), which translates to £37.49 here (incl. VAT).
■ He also lists two books by Conrad Carlberg, and published by Que: *Managing Data with Excel*, ISBN: 0-7897-0382-3 and *Business Analysis with Excel*, ISBN: 0-7897-0385-8.
■ Among his recommendations for VBA beginners is one I bought and use myself: *Developing Excel 95 Solutions* by Eric Wells (no relation), published by Microsoft Press, ISBN: 1-55615-893-9, 864 pages. It includes a 3.5in disk and costs £37.49 here.
■ I must look out for another of his suggestions because it's only $9.95 in the US. It's called *Microsoft Excel Visual Basic Macros Made Easy*, by Bob Flanagan, but he gives no other details.

Andy and Malcolm are in very different situations but from a spreadsheet point-of-view, their problems have a common thread. What they're both trying to do is prepare a holiday schedule with many variables and a number of constraints.

We know what's best

Malcolm had posted his problem to several newsgroups and a chap called Euan in New Zealand said that such preferred assignment schedule riddles are technically known as linear optimising problems. He raved about an Excel add-in called What'sBest! — I love these worldwide conversations, and to think I remember the time when you had to make a booking to talk to relatives overseas at busy times!

What'sBest! (Fig 3) is a well-established product which I hadn't used it before so I tracked down the US publisher and UK distributor and was sent the professional version of What'sBest! Release 2.1. This release supports Excel 3, 4, and 5 (and Lotus 1-2-3 Release 4 and 5) running in Windows 3.x, Windows 95, or Windows NT. Excel 7 is supported only on Windows 95. The product is also available for Quatro Pro and Symphony.

Delivered on two 3.5in disks, What'sBest! is easy to install and use. There is an excellent, well-indexed 256-page manual with a good tutorial. There are lots of sample worksheets covering everything from bond portfolio optimisation to hogfeed mixes and truck loading. There are three staff-scheduling templates included.

This spreadsheet add-in can solve both linear and non-linear optimisation problems and can be restricted to whole units so you don't schedule 3.29 persons or 4.9 days.

You start in the way I always recommend: set up a worksheet with the answers missing, then go through the three steps called ABCs. A is for Adjustable Cells, B stands for Best, and C is for Constraints. The Adjustable Cells are those where you would enter your guesses if you were trying to solve your problem manually. In some industries they are called decision variables. These are the cells with which What'sBest! can play about, to come up with the solution. The Best is the goal of your solution. It might be to maximise or minimise an adjustable cell. No Best is needed when goal-seeking.

The Constraints are the limitations. A Constraint might be: that the number of employees scheduled for each day must be at least equal to the staffing needs for that day; or that preference for days off is ranked by seniority; or that nobody must work more than five consecutive days.

The number of variables and constraints which What'sBest! can handle depends on the version. The bottom-of-the-line personal version can handle 400 variables and 200 constraints. It needs only 640K of memory under DOS, or 4Mb under Windows. The middle-level professional version copes with 4,000 variables, and 2,000 constraints and needs 8Mb of memory under Windows. The high-end extended version copes with 32,000 variables and 16,000 constraints and

On the CD

This month, in the *Hands On Spreadsheets* section of our cover-mounted CD, there is a worksheet template for printing cheques. The clever stuff in it, which automatically writes the amount in words after you've entered it in figures, was written by veteran contributor Shane Devonshire of Walnut Creek, California. I had to anglicise much of his design for the cheque, though. Although I lived in the US for many years, I'd forgotten how much their banking
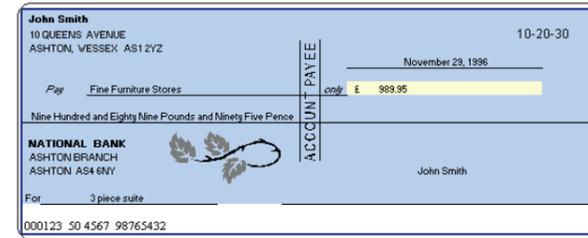
conventions differ. They don't just spell cheque differently and use dollar signs, the wording and reference number arrangements are not the same. It is easy to substitute your own personal and bank details in this template, so you can print custom-designed cheques. I've provided the file in two formats: CHEQUE.xls and CHEQUE.wk4. The 1-2-3 file doesn't have the drawing objects which are in the Excel version but all the formulas work.

needs 32Mb of memory under Windows. What'sBest! writes it's own formulas but you can prefer. You can also use macros written in Excel 4 or 1-2-3-macro languages or Visual Basic to execute any of its commands. Simple folk like me can start by selecting one of the sample files and modifying it to fit. What'sBest! can also be used by experienced programmers to build a fancy application.

■ If you would like to suggest alternative ways of solving Andy's or Malcolm's problem, do let me know. An elegant solution (emailed as an attached file or sent on a disk) that works straight away for me, could win you a book token.

Shortcuts follow-up

In the *EXCELlent Shortcuts* section of my September issue column, I mentioned that the F4 key will change a cell reference from relative to absolute. But reader Chris Vivian expands on this by pointing out that F4 actually toggles and will cycle through relative column and absolute row, absolute column and relative row, and so on.

Chris additionally reminds us that you can check how Excel utilises all the function keys by choosing Help, Topics, Index and then entering Shortcut. Then choose built-in keyboard shortcuts. This offers eight categories of shortcuts.

The last option is Function Keys. This not only shows what the function keys do by themselves, but also in combination with the Shift, Ctrl, Ctrl+Shift and Alt+Shift keys. For example, F1 is Help but Alt+Shift+F1 inserts a new worksheet. F3, F6 and F12 each offer four different functions if you can remember the combinations.

Also in the September issue, I described how to make a pick list in Excel 7. One reader, Kelvin Syrett, asked how he could construct one in Excel 4, so I drew his attention to the Dialogue Editor which comes as an accessory with Excel 4. A list box is one of the many features you can create with this editor. To find out how it is done, see Chapter 8 of Book 2 of the *Excel 4 User's Guide*.
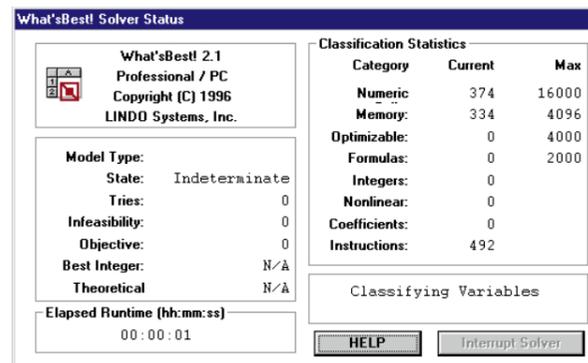
Fig 3 What's Best! offers this continually updated report of what's going on while it calculates

EXCELlent shortcuts and longshots

1. UNDERCOVER To hide data in cells, select the cells. Choose Format, Cells, Number. In the Category box, click Custom. In the Type box, enter three semicolons (;;;). The data contained in the cells will appear in the formula bar, or in a cell if you press F2 and edit within the cell. The data in the cell will not be printed.

2. GETTING A DATE To find a date in a range of dates that's closest to a specified date, you can use this formula:

```
{=IF(ISNA(VLOOKUP(Date-
MIN(ABS(Table-Date)),
Table,1,0)),VLOOKUP(Date+MIN(ABS(
Table-Date)),Table,1,0),
VLOOKUP(Date-MIN(ABS(Table-
Date)),Table,1,0))}
```

It assumes that the range of dates has been named Table and the cell containing the specified date is named Date. The curly brackets are not entered. This is an array formula and the brackets will appear when you press Ctrl+Shift+Enter.

3. WHO'S THERE? It is easy to compare two columns of text and find matches and mismatches. Say you have a list of names in the range, A9 to A14, and another in C9 to C14. In cell D9 put:

```
{=IF(OR(A9=$C$9:$C$14),A9,"")}
```

Again, this is entered as an array formula. Then copy this formula down to D14. Column D will then show any names which appear both in columns A and C. In cell E9 enter another array formula:

```
{=IF(AND(A9<>$C$9:$C$14),A9,"")}
```

Copy this down to E14. Column E will then display any names in column A which are not in column C. Finally, in F9 the array formula is:

```
{=IF(AND(C9<>$A$9:$A$14),C9,"")}
```

Copy this down to F14. Column F will then display any names which appear in column C but not in column A.

(Tips 2 and 3 are courtesy of Shane Devonshire.)