

THE DOUGHNUT GRINDER'S LOGBOOK

Here's an AppleWorks spreadsheet template for cyclists, joggers and swimmers that helps you to monitor your progress.

I hate to exercise. I hate to be fat. I hate being fat worse than I hate exercising, but it's close. To provide some incentive to get on my bicycle (affectionately known as the Doughnut Grinder) and ride, I keep track of my times and distances using my Apple IIc and AppleWorks. Handling time with a spreadsheet can present some problems, because the spreadsheet wants things based on 10 and powers of 10, not 60 seconds equals 1 minute and 60 minutes equals 1 hour.

The template described here can also be used to keep track of your jogging progress. Figure 1 shows the screen after the labels have been entered and Figure 2 shows the screen with some sample values entered. The template formulas convert from minutes and seconds to decimal values and vice versa. The same formulas could be used to convert hours and minutes for applications such as payroll spreadsheets.

Only a rudimentary knowledge of AppleWorks is assumed here. You can plug the formulas into the spreadsheet and use them without understanding how they work. That's the beauty of a template — the hard work's done!

To use the completed template, just enter the number of miles you pedal or run for a particular day in column B and the time it took in column C. Separate the hours from the minutes with a single decimal point. The spreadsheet will calculate the average speed for the month, the average miles, the total miles, the average time (expressed as a decimal) and the average time (expressed as minutes and seconds). See Figure 2 for a sample screen.

ENTERING THE TEMPLATE

First, boot up AppleWorks and from the Add Files menu, choose to make a new file for the spreadsheet. Then choose the From

Scratch option and give your file a name such as Cycle.Logbook. You may want to keep one blank master copy on your disk. Then when you start each month, you can load in a copy of the master and rename it with Open-Apple-N to a name that includes the month.

Labels

Start by entering the labels as shown in Figure 1. For proper alignment, the numbers in column A should be entered as labels rather than values. This is accomplished by starting each entry with

1	DATE	TIME	AVG. DAILY
2	MILES	(mm:ss)	SPEED
3			
4:1			
5:2			AVERAGE SPEED:
6:3			
7:4			AVERAGE MILES:
8:5			
9:6			TOTAL MILES:
10:7			
11:8			AVG. TIME (mm:dec):
12:9			
13:10			AVG. TIME (mm:ss):
14:11			
15:12			
16:13			
17:14			
18:15			

FIGURE 1: Screen Dump Showing Label Entries

a quotation mark (""). Note that these continue all the way up to 31 in cell A34. The expressions in parentheses in cells C2, G11 and G13 also must be entered with a leading quote.

Note that some of the labels span two adjacent cells. To make these appear properly on the spreadsheet, stop typing when you reach the right edge of the first cell. Then press the Right-Arrow key and continue with the rest of the label in the cell just to the right of the first one. If you keep typing when you reach the edge of the first cell, the remaining text will go in the first cell, but it

The same formula calculates hours and minutes...

won't appear on the screen with the default column widths.

Since AppleWorks doesn't allow initial entries with leading spaces, you will have to use the Open-Apple-U command later and insert the spaces in each field that requires them. For the dates in column A, use four spaces before numbers 1-9 and three before the others.

Formulas

The formulas in column D are all essentially the same. The only differences are dependent on the row number. All you have to do is type one formula in cell D4 (D4: indicates the cell in which the formula goes):

D4: `+@IF(B4 > 0, B4 / (@INT(C4) + (C4 - @INT(C4) / .6)) * 60, 0)`

Then use the Copy command, following these steps:

1. Move the cursor to cell D4.
2. Press Open-Apple-C and select Within Worksheet.
3. Press Return to indicate the single cell D4 as the source.
4. Move the cursor to D5 and press the Period key to indicate the beginning of the destination range.
5. Move the cursor to D34 and press Return to indicate the end of the destination range.
6. You are then asked several times about references to cells B4 and C4. Specify that all of these references should be relative.

Then enter the formulas for cells H5, H7, H9, H11 and H13:

H5: `+@IF(@COUNT(B4...B34) > 0, @SUM(D4...D34) / @COUNT(B4...B34), 0)`

H7: `+@IF(@COUNT(B4...B34) > 0, @AVG(B4...B34), 0)`

H9: `+@SUM(B4...B34)`

H11: `+@IF(@COUNT(B4...B34) > 0, (H7/H5)*60, 0)`

H13: `+@IF(@COUNT(B4...B34) > 0, @INT(H11) + ((H11 - @INT(H11)) * .6), 0)`

Finally, press Open-Apple-V and choose Value Format, Fixed and two decimal places. This will make all of the values on the spreadsheet appear with two decimal places.

Save the completed spreadsheet to a ProDOS formatted disk by selecting Save Desktop Files from the main menu or by using Open-Apple-S while the spreadsheet is displayed.

HOW THE SPREADSHEET WORKS

The key to the operation of this spreadsheet is the conversion between times in minutes.seconds format and times in decimal format. The minutes.seconds format times are easier to enter and don't involve any preliminary mental calculations. AppleWorks, however, requires all its values in decimal numbers. As part of calculating the average daily speed, the formulas in column D convert the minutes.seconds values entered in column C to decimal values. The formula in cell H13 converts the average time to minutes.seconds

format from the decimal format in cell H11. All six different formulas are described in detail below.

D4: `+@IF(B4 > 0, B4 / (@INT(C4) + (C4 - @INT(C4) / .6)) * 60, 0)`

Whew! — looks rugged. But, really, it's not. In plain English, this formula says, "If the value in cell B4 is greater than zero [`+@IF(B4 > 0)`], divide the value in cell B4 [`B4/`] by the time [`(@INT(C4) + (C4 - @INT(C4) / .6))`]; otherwise, put a zero in the cell." Was that hard? Well, maybe the time formulation was. Let's look at that part a little more closely.

Our goal here is to separate the minutes from the seconds, convert the seconds to a decimal part of a minute, and then add the minutes back in so that the result can be used in other calculations. The built-in AppleWorks function, INT, is the key to converting the time to a usable format. The time portion of the formula says to take the integer part (everything to the left of the decimal point) of the number in cell C4 and add it to what follows. Since the time is entered as *mm.ss*, the minutes are already separated from the seconds, so they can be manipulated. Next, we tell it to take the entire number in C4 and subtract the integer part from it. This separates out the seconds. For example:

$$mm.ss - (integer\ part) = 0.ss$$

$$35.45 - 35 = 0.45$$

Now you need to convert the seconds to decimal format.

To calculate minutes in decimal format, we ask, "What part of a minute is *x* seconds?" or, since a minute is 60 seconds, "What is the ratio of *x* seconds to 60 seconds?" For example, 30 seconds divided by 60 seconds is 0.5, which is 0.5 minutes. Why does the

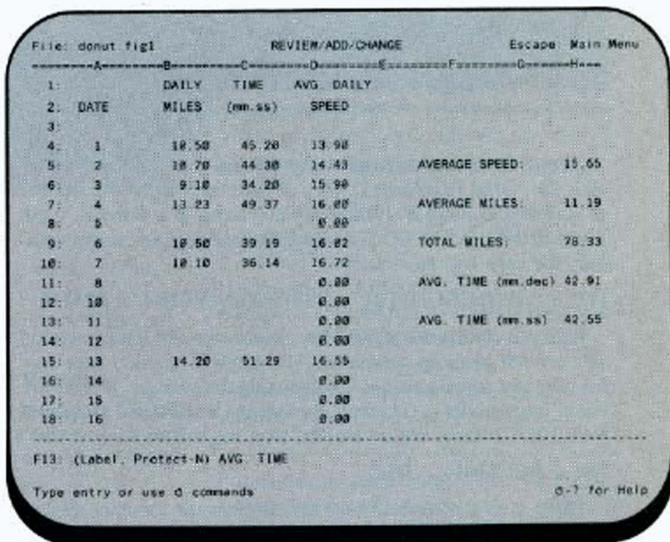


FIGURE 2: Screen Dump Showing Sample Results

formula say, "divided by 0.6" (`/ .6`), and not 60? The spreadsheet was told to take the integer from the entire number, which leaves us with a decimal number.

Look at the example above again. The answer is 0.45, which is 45 seconds, but AppleWorks doesn't know that. Forty-five divided by sixty is the same as 0.45 divided by 0.6 which equals 0.75. That's what we're interested in; forty-five seconds is 3/4 of a minute, or 0.75 minutes. Now add that to the integer part so cleverly broken off earlier and our conversion is complete.

"If the conversion is complete," you ask, "what's that '*60' for? The formula calculated how many miles were covered in one minute. You still need to find out how many miles you covered in 60 minutes, or 1 hour, so your answer will be in miles per hour. The '*60' does that. The ',0' tells AppleWorks what to put in the cell if B4 is empty.

Except for the references to cells in columns B and C, the rest of the formulas in column D are exactly the same as the one in cell D4.

Take heart, folks. The worst is over!

H5: `+@IF(@COUNT(B4...B34)>0,@SUM(D4...D34)/@COUNT(B4...B34),0)`

This formula calculates the average of your daily speeds. AppleWorks does have a function that will compute an average (@AVG), and if you plan to ride your bicycle every day without fail, you can use it. But if you're like me and occasionally skip a day or two (or three) for business trips or illness (real or imagined), you can't use AVG because the function treats a zero as an entry. This means that if you rode 23 days of the 31 in the month, your speed would be divided by 31, not 23. Even if you rode all 30 days in the month of June, your speed would be divided by 31. This formula avoids those problems.

I told the spreadsheet to put a zero in those cells if column B had no entry. Why not just drop that command? Because when AppleWorks tries to use those empty B and C column cells in the calculations for column D, it doesn't know how to handle the blank cell, and you get an ERROR message instead of a value.

Handling time with a spreadsheet can present some problems, because the spreadsheet wants things based on 10...

If there are one or more entries in column B, the formula above adds the entries in column D, then divides the total by the number of entries in column B. If there are no entries in column B, it put a zero in the cell. The COUNT function does not count empty cells, only the cells that have entries.

H7: `+@IF(@COUNT(B4...B34)>0,@AVG(B4...B34),0)`

Here you can use the @AVG function because the days you don't ride are left blank in column B. The @AVG function uses only the cells that contain entries in calculating the average. If there are one or more entries in column B, the average is calculated and placed in cell H7. If there are no entries, then the cell receives a zero.

H9: `+@SUM(B4...B34)`

Totaling all the entries in column B gives the monthly total to date. (Have you noticed that these formulas are getting easier?)

H11: `+@IF(@COUNT(B4...B34)>0,(H7/H5)*60,0)`

The average daily mileage divided by the average daily speed gives the average daily time in minutes and decimal minutes. By now, you should be able to figure out why the '*60' is there. (Hint: Did we compute miles per hour or miles per minute?)

H13: `+@IF(@COUNT(B4...B34)>0,@INT(H11)+((H11-@INT(H11))*6),0)`

In this formula, the operation of the first formula is reversed: it converts minutes and decimal seconds to minutes and seconds.

BELLS AND WHISTLES

With all the formulas installed, the spreadsheet will work just fine. But AppleWorks has one more feature you may want to install in your spreadsheet: protection.

Move to cell A1. Press Open-Apple-L, choose Columns, highlight column A, choose Protection, and then respond to the Allow prompt with Nothing. This will prevent you from accidentally writing over your dates. As you may have noticed, you can protect single entries, rows, columns or blocks. I protected those areas where the results would be displayed, such as the block of columns F-H, rows 5-13, and column D. AppleWorks will now stop me from doing something like entering an average speed of 43.45 when I meant to put it in the TIME column, which would mean that I would have to retype one of those formulas. Notice that protection means that AppleWorks can write in that space, but you can't.

You can also use protection to allow only values (i.e., no labels) to be inserted in a cell. You can use the Block choice to protect cells B4-B33 and C4-C33.

TIPS

I finished my template and named it Cycle.Logbook and then saved it. Next, I called it up and used Open-Apple-N to change the name of the file to BIKE.current month. I make my entries each day and when the month is over, I call up the template and rename it. Once your master copy is working, you can exit AppleWorks and lock the file from FILER or the System Utilities. AppleWorks will then be unable to accidentally overwrite your master file.

AppleWorks does its figuring by columns or rows. For instance, it looks at row 1 and does any calculations in that row, then it does the calculations in row 2, and so on. Or it looks at column 1, then column 2, and so on, until it finishes the entire spreadsheet. The default setting is Rows. For this reason, once you've made an entry, "ERROR" will be displayed in the summary area because AppleWorks doesn't have enough information to do the work. Once you have made all the daily entries, use Open-Apple-K to make the sheet recalculate everything. The ERROR messages will disappear.

You can do one of two things to make the sheet work more smoothly. You can press Open-Apple-V, and choose Recalculate, Order, and then Columns. Now, when you enter DAILY miles, you see the ERROR message in columns D and H, but when you enter TIME, all the ERROR messages are replaced by the correct results. I prefer this solution.

Your other choice is to leave the order of calculation alone and move the summary area to the bottom of the spreadsheet. This solves the recalculation problem, but you have to scroll the sheet to see the results. On the other hand, if you're not doing so well, your dismal record won't be right up front.

MODIFICATIONS

Now that you know how a spreadsheet can handle time calculations, you're not limited to the bicycle logbook. The same formula calculates hours and minutes; just relabel the column headings. You can't enter hours, minutes and seconds in the same format, since AppleWorks won't accept two decimal points in a value field. But you can make a separate column for hours (or seconds) and tell it to add the two columns. If I ever get to the point where I can ride for more than an hour, I'll just convert the hours to minutes myself and then enter them (e.g., 65 minutes). Remember, the spreadsheet doesn't know that there are 60 minutes in an hour, so it will happily accept 95.45 or 256.32 as an entry.

The software's ready. Now all you need is the will power — and pedal power — to work up the entries!

