



## hp calculators

HP 20b Using the RPN stack to solve problems efficiently

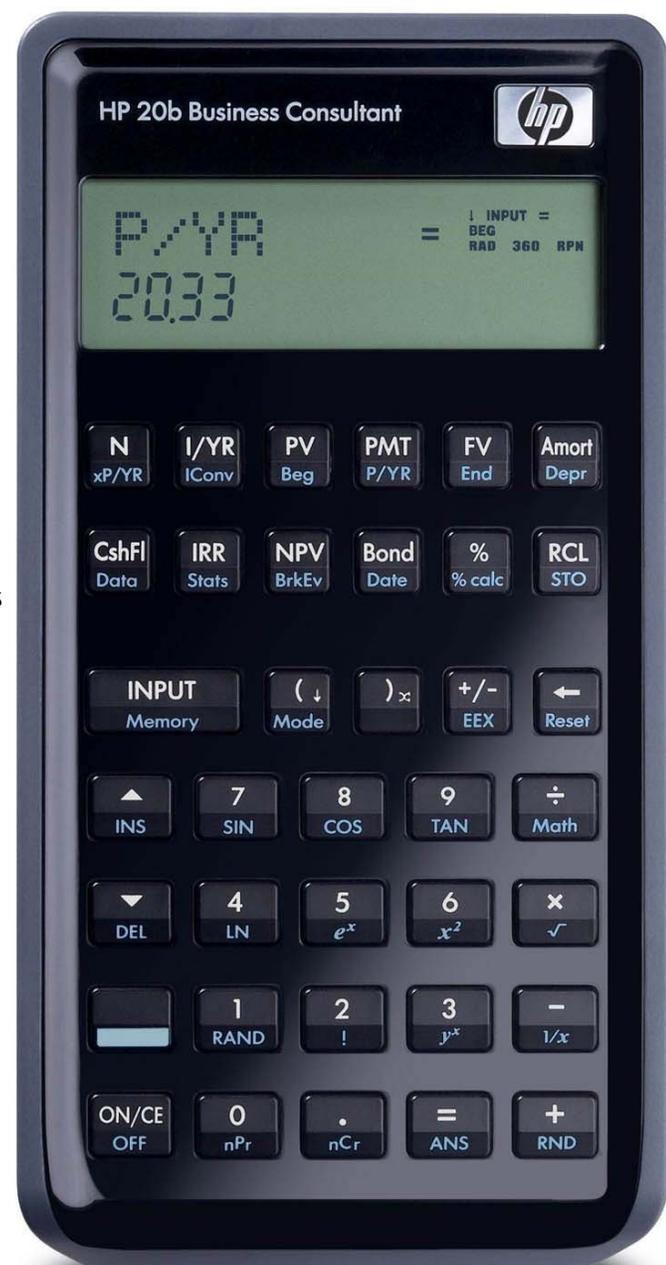
The benefits of understanding RPN stack operations

Understanding the HP 20b RPN stack operation in RPN mode

Viewing and reordering stack-register contents

Using ANS contents in chain calculations

Practice solving problems using stack registers



### The benefits of understanding RPN stack operations

RPN stands for Reverse Polish Notation and is one of two operating modes available on the HP 20b. Mastering RPN leads to an enhanced performance when using the calculator. The first step to master RPN usage is to know all of its available resources.

To put the HP 20b into RPN mode, press  **Mode**, then press  five times. The display should show either Chain, Algebraic or RPN. If it does not show RPN and you wish to put the HP 20b into RPN mode, press  until the display shows RPN (which would only take one or two times). To exit this menu, press .

When RPN mode is active, the RPN annunciator is visible in the display as shown in figure 2 below.

### Understanding the HP 20b stack operation in RPN mode

When in normal, 'run' mode, every operation performed in the HP 20b uses the display contents or places results on it. The display always shows the contents of the **X-register**. A register is a predefined place in the calculator memory that is able to hold a formatted number with a twelve-digit mantissa and a three-digit exponent of ten. The X-register is one of five registers that form the RPN stack, represented in Figure 1 with all contents cleared to zero.

Stack registers

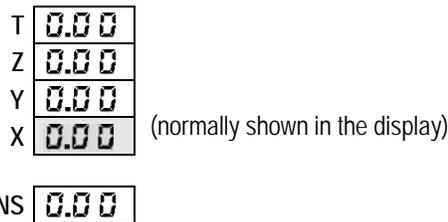


Figure 1

Anytime a number is keyed in, the X-register is updated to hold this number. It is then available for use.

**Example 1:** What is the most common keystroke sequence to compute the reciprocal of 7.2? What registers in the stack are used?

**Solution:** The most common keystroke sequence to compute the reciprocal of 7.2 is:

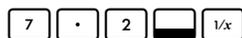


Figure 2

In this case, simply typing in the number and pressing the related key are enough to compute the resulting value. There are other functions that use only the X-register contents, like  **!** or  **TAN**. These are the 'one-number functions', and when they are pressed, the calculator proceeds with the following operation sequence:

HP 20b Using the RPN stack to solve problems efficiently

---

1. take a copy of the X-register contents and put it in the ANS register;
2. replace actual X-register contents with the resulting value.

If a second number needs to be typed in right after the first one to be used with it, they must be separated with the  key. For example, to add 3 and 5 together, you would press    . Note: The  key can also be used like the  key. Long time HP users will probably feel more comfortable using the double-wide key.

Example 2: What is the most common keystroke sequence to enter 2.33 and 4.5 and add them to each other? What registers in the stack are used?

Solution: The most common keystroke sequence to add 2.33 to 4.5 is:



Figure 3

In fact, all of the stack registers have their contents changed during this single addition. However, the real question is that since both numbers must "exist" in the calculator memory before  is pressed, "Where do they exist?" When  is pressed, the calculator proceeds with the following operation sequence:

1. take a copy of the X-register contents and put it in ANS register;
2. take Y-register contents and add to X-register contents;
3. replace actual X-register contents with this resulting value;
4. "drop" T- and Z-register contents respectively to Z- and Y-registers;
5. maintain a copy of T-register contents in T.

**NOTE:** On the HP 20b, the  key terminates digit entry and places the keyed value into the X register. It does not copy this value into the Y-register the way previous RPN calculator implementations have done. In this sense, the HP 20b operates more like the graphical RPL calculators. Therefore, when you want to evaluate  $4 + 5$  using these keystrokes:    , the HP 20b would place the 4 into the X-register when  is pressed, but only copy this 4 into the Y-register when the 5 is keyed and an operation is performed on the 5. In most normal situations, this will not cause any issues, but a long time RPN user should be aware of this behavior.

To have the HP 20b copy the value from the X-register to the Y-register, you should press  twice after keying a number.

**Viewing and reordering stack registers contents**

Example 3: How would you fill the T, Z, Y and X stack registers with the numbers 44, 33, 22 and 11, respectively?

Solution: The most common keystroke sequence to fill the stack registers with these numbers is:

The stack registers contents would be updated according to the diagrams shown below:

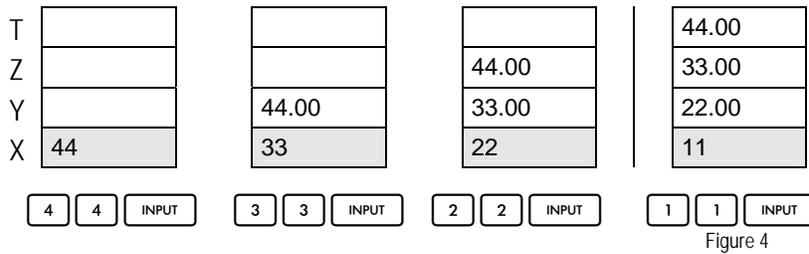


Figure 4

Under certain circumstances, keeping track of whatever is in each stack register is not practical for daily, quick computations. To help viewing the stack contents, three functions are provided:  $\leftarrow$  (swaps the contents of the X and Y registers) and  $\downarrow$  and  $\uparrow$  (both keys roll-down all stack registers contents) and  $\uparrow$  (roll-up all stack register contents). Keeping previous contents as they are, watch how the following keystrokes change the stack:

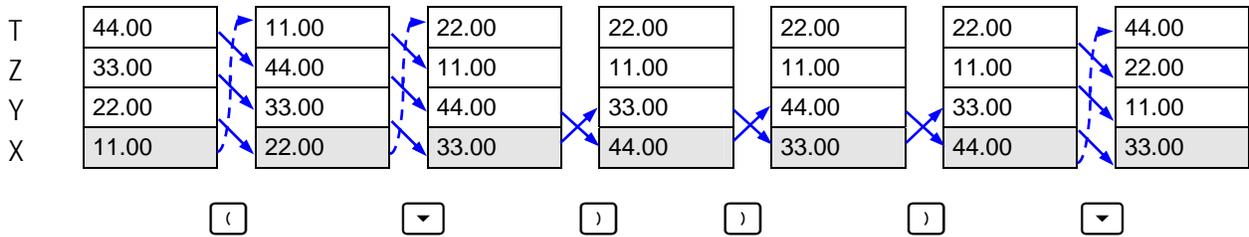


Figure 5

### Using ANS contents in chain calculations

The ANS-register is automatically updated anytime the X-register contents are changed as the result of a calculation. In this case, the ANS-register is loaded with a copy of the last value in X prior to executing a function. Whenever possible, using the ANS-register contents in chain calculations avoids having to type in numbers or intermediate results again. The ANS-register's contents can be retrieved to the X-register anytime it is needed with the sequence  $\leftarrow$  ANS.

**Example 4:** What is the shortest keystroke sequence to find y value in the following expression given  $x=3.4567$ ?

$$y = \sqrt{x} + 2 \times \ln(x)$$

Figure 6

**Solution:** The shortest keystroke sequence to compute y is:



Figure 7

When  $\leftarrow$   $\sqrt{\phantom{x}}$  is pressed after 3.4567 is keyed in, the ANS-register is loaded with 3.4567, a copy of X-register contents, before the square root is applied. Right after  $\leftarrow$   $\sqrt{\phantom{x}}$ ,  $\leftarrow$  ANS retrieves ANS-register contents back to the X-register so LN(x) can be calculated. What must be remembered is that  $\leftarrow$  ANS also pushes up the stack contents. The sequence  $\leftarrow$  LN 2  $\times$  computes the second part of the right side of the equation, once the original value of X has been brought back from the ANS-register.

HP 20b Using the RPN stack to solve problems efficiently

---

Using the stack registers efficiently does not mean to keep track of all stack register contents all the time. Instead keeping track of what is happening, like which registers have their contents being used, is helpful.

Practice solving problems with stack registers

Example 5: Now what is a short keystroke sequence to find  $y$  value if the previous expression is changed for the one below, given  $x=3.4567$ ?

$$y = \sqrt{x} + 2 \times \ln(x) - \frac{1}{x}$$

Figure 8

Solution: One of the shortest keystroke sequences to compute  $y$  in the expression is:



Figure 9

These examples only begin to scratch the surface of the possibilities that exist when using RPN.