

To this the king replied, "But, generous gentleman, this is a prosperous kingdom; surely I can do more for you than that!" But the gentleman was equally insistent.

P.S. The kingdom produced about one billion (1,000,000,000) stalks of wheat (W) annually and the chessboard had 64 squares (S) to be filled.

Was the gentleman really being generous?

The program calculates the amount of wheat to be placed on each square in succession in R_2 and keeps track of the number of the square in R_1 . If more than one billion stalks of wheat have to be supplied for a given square, the program halts and displays the number of the square at which it surpasses one billion. If the 64 chess squares can be filled without depleting the kingdom's supply, the program halts and displays the number of stalks of wheat that need to be paid.

Switch to W/PRGM mode, press **f** **PRGM** to clear program memory, and key in the following list of keys.

Keys	Comments	Keys	Comments
LBL }	Beginning of program.	6	
A }		4	
1 }		9 x=y	Compare square number to 64.
STO 1 }	Initialize R_1 and R_2 to take care of 1st square.	RCL 2 }	If square number equals 64, display amount and stop.
STO 2 }		RTN	
LBL }	Beginning of repeat.	RCL 2 }	Otherwise, compare amount to 1 billion (1×10^9).
1 }		EEX	
2 }		9 x>y	Branch if one billion is greater than amount.
STO }	Calculate amount.	GTO }	
x }		1 }	
2 }		RCL 1	Otherwise, display square number and stop.
RCL 1 }	Increment square number.	RTN	
1 }			
+			
STO 1			

Now switch back to RUN mode and run your program. After several seconds the calculator should display:

31.00

Over one billion stalks of wheat have to be placed on the 31st square!

(To find the exact amount on that square press **RCL** **2**.) To calculate the amount on the 64th square, press **2** **ENTER** **6** **4** **9** **y^x**. Needless to say, the generous gentleman was executed!

The second program calculates the arc sine of an input value x (x must be within the limits of -1 and $+1$.) The program tests the resulting angle, and if it is negative or zero, adds 360 degrees to it to make the angle positive.

Switch back to W/PRGM mode, press **f** **PRGM**, and key in the program now.

Keys	Comments	Keys	Comments
LBL }	Beginning of program.	9 x≤y	Test the angle.
D }		3 }	These two steps are skipped if angle is positive.
f¹ }	Calculates the arc sine.	6 }	
SIN }		0 }	Adds 0 or 360 to assure that the angle is positive.
0 }	Puts 0 in X.	+	
9 x≤y	Exchanges 0 and arc sine.	RTN	

If the angle is positive, **3** and **6** are skipped and zero is added to the angle. Otherwise, 360 is added to the angle. Let's try a problem.