Backtracking Euclid’s Algorithm to find Modular Multiplicative Inverse, MMI

Suppose we want to know the MMI of 11 MOD 60. This means we want to know the number that when multiplied by 11 gives a number that is one larger than a multiple of 60. The answer in this case is pretty easy to see because we know that 11*11 = 121 = 2*60 + 1 so that 11 is the MMI of 11 MOD 60. How do we find the MMI without “seeing” the answer?

Example 1. We know that there will be a MMI because 11 and 60 are coprime. Nonetheless, following Euclid’s Algorithm we see that 11 and 60 have a GCD of 1 in 3 iterations as shown in the left half of the table below. The right half simply rewrites each iteration (except the last) solving for the remainder (noting Integer*Smaller is now subtracted).

Larger	=	Integer	*	Smaller	+	Remainder	Iteration, ***k***	Remainder	=	Larger	–	Integer	*	Smaller
60	=	5	*	11	+	5	1	**5**	=	60	–	5	*	11
11	=	2	*	5	+	1	2	1	=	11	–	2	*	**5**
5	=	5	*	1	+	0	3							

To find the MMI, start at the bottom and work upward using the equations on the right hand side.

Substitute the 1st into the 2nd in place of **5** we have: 1 = 11 – 2*(**60 – 5*11**)

Distributing: 1 = 11 – 2*60 + 2*5*11

Regrouping we have 1 as a multiple of 11 and 60: 1 = – 2*60 + (1 + 2*5)*11 = **11***11 – 2*60 .

This says that 11 is the MMI of 11 MOD 60.

Example 2. A more challenging question is: what is the MMI of 11 MOD 52? This is not readily apparent but is easily found by backtracking Euclid as was done above.

Larger	=	Integer	*	Smaller	+	Remainder	Iteration, ***k***	Remainder	=	Larger	–	Integer	*	Smaller
52	=	4	*	11	+	8	1	**8**	=	52	–	4	*	11
11	=	1	*	8	+	3	2	**3**	=	11	–	1	*	**8**
8	=	2	*	3	+	2	3	**2**	=	8	–	2	*	**3**
3	=	1	*	2	+	1	4	1	=	3	–	1	*	**2**
2	=	2	*	1	+	0	5							

To find the MMI, start at the bottom and work upward on the right hand side.

Substitute the 3rd into the 4th in place of **2** and we have: 1 = 3 – 1*(**8 – 2*3**)

Distributing: 1 = 3 – 1*8 + 1*2*3

Regrouping we have 1 as a multiple of 3 and 8: 1 = – 8 + (1 + 2)*3 = – 8 + 3***3**.

Substitute the 2nd in place of **3** and we have: 1 = – 8 + 3*(**11 – 1*8**)

Distributing: 1 = – 8 + 3*11 – 3*8

Regrouping we have 1 as a multiple of 8 and 11: 1 = 3*11 – (1 + 3)*8 = 3*11 – 4***8**

Substitute the 1st into in place of **8** and we have: 1 = 3*11 – 4*(**52 – 4*11**)

Distributing: 1 = 3*11 – 4*52 + 4*4*11

Regrouping we have 1 as a multiple of 11 and 52: 1 = – 4*52 + (3 + 16)*11 = **19***11 – 4*52 .

This says that 19 is the MMI of 11 MOD 52.

When Euclid has remainder 0 in an odd Iteration, *k*. You may have noticed that these two examples both achieve a zero remainder in an odd iteration (***k*** = 3 and ***k*** = 5). When this happens, the backtracking process produces a positive value as a multiple of the Smaller number. That multiple is the MMI of the Smaller number MOD Larger number. The only difference is the number of backward substitutions necessary to end up writing 1 as a function of the Larger and Smaller numbers. This will occur after ***k***-2 substitutions.

When Euclid has remainder 0 in an even Iteration, *k*. When the Euclidean Algorithm takes an even number of iterations to achieve remainder 0 the result of the backtracking process produces negative numbers as the multiple of the Smaller number. In this instance, we must do one more step to find the MMI of ***a*** MOD ***b*** as we see in this example.

Example 3. What is the MMI of 11 MOD 59? This is not readily apparent but is also easily found by backtracking Euclid.

Larger	=	Integer	*	Smaller	+	Remainder	Iteration, ***k***	Remainder	=	Larger	–	Integer	*	Smaller
59	=	5	*	11	+	4	1	**4**	=	59	–	5	*	11
11	=	2	*	4	+	3	2	**3**	=	11	–	2	*	**4**
4	=	1	*	3	+	1	3	1	=	4	–	1	*	**3**
3	=	3	*	1	+	0	4							

To find the MMI, start at the bottom and work upward using the equations on the right hand side.

Substitute the 2^{nd} into the 3^{rd} in place of **3** we have: $1 = 4 – 1*(\mathbf{11 – 2*4})$

Distributing: $1 = 4 – 1*11 + 1*2*4$

Regrouping we have 1 as a multiple of 4 and 11: $1 = – 11 + (1 + 2)*4 = – 11 + 3*\mathbf{4}$.

Substitute the 1^{st} in place of **4** and we have: $1 = – 11 + 3*(\mathbf{59 – 5*11})$

Distributing: $1 = – 11 + 3*59 – 3*5*11$

Regrouping we have 1 as a multiple of 11 and 59: $1 = 3*59 – (1 + 15)*11 = 3*59 – 16*11$.

So, -16 is ***an*** MMI of 11 MOD 59.

But -16 is not a number between 1 and 58 so it is not ***THE*** MMI of 11 MOD 59.

The MMI is defined as a positive number less than the Larger number (the modular base) that, when multiplied by the Smaller number, produces a result that is 1 more than a multiple of the Larger number.

The Extra Step. To get the MMI, add 59 (the Larger number) to the result. In this instance, 43 = 59 – 16.

43 is the MMI of 11 MOD 59.

Here is a check: 43*11 = 473. But 8*59 = 472, so: 1 = 43*11 – 8*59, just as required by MMI.

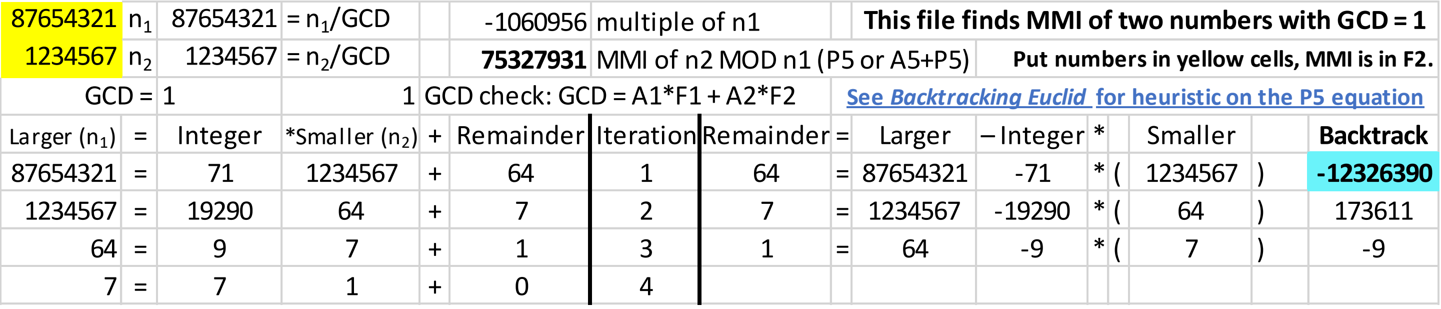
In words: 43 times 11 produces a number that is 1 more than a multiple of 59, just as required by the definition of MMI.

An automated version in Excel. This backtracking process can be set up in Excel, much as Euclid's Algorithm for finding the GCD was shown in the *Euclid's Algorithm in Excel* file (**PwP** File MA.2.b.i). The ***Backtrack equation*** required for this process (cell P5 in *Finding MMI from Euclid.xlsx*) is shown at the bottom of the page without formal discussion of why it works but the reason it works is that it mimics the iterations discussed above. When GCD(n_1,n_2) > 1, a "quasi-MMI" is calculated as long as the Smaller number is not a multiple of the Larger number (i.e., as long as GCD(n_1,n_2) < MIN(n_1,n_2)).

A non-*PwP* example. **Q:** What is the MMI of 1,234,567 MOD 87,654,321? **A:** 75,327,931.

The image below shows the Euclidean Algorithm to the left, and the backtracking to the right. The far right numbers are the MMI check with the top = n_2*MMI, and bottom = n_1*-1060956.

MMI Check
92,997,377,790,877
-92,997,377,790,876



87654321	n_1	87654321	= n_1/GCD		-1060956	multiple of n1			**This file finds MMI of two numbers with GCD = 1**						
1234567	n_2	1234567	= n_2/GCD		**75327931**	MMI of n2 MOD n1 (P5 or A5+P5)			**Put numbers in yellow cells, MMI is in F2.**						
GCD =	1		1	GCD check: GCD = A1*F1 + A2*F2					See *Backtracking Euclid* for heuristic on the P5 equation						
Larger (n_1)	=	Integer	*Smaller (n_2)	+	Remainder	Iteration	Remainder	=	Larger	– Integer	*		Smaller		**Backtrack**
87654321	=	71	1234567	+	64	1	64	=	87654321	-71	*	(	1234567	)	**-12326390**
1234567	=	19290	64	+	7	2	7	=	1234567	-19290	*	(	64	)	173611
64	=	9	7	+	1	3	1	=	64	-9	*	(	7	)	-9
7	=	7	1	+	0	4									

Backtrack equation in P5: =IF(F5="","",IF(F5=0,"",IF(F6=0,K5,IF(F7=0,K6*K5+1,IF(F8=0,P6*K5+K7,IF(F8>0,K5*P6+P7))))))