00063D

Program Description I

Program Title FACTORS AND PRIMES

Contributor's Name HEWLETT-PACKARD COMPANY
Corvallis Division
Address 1000 N.E. Circle Boulevard
City Corvallis, OR 97330
State **Zip Code**

Program Description, Equations, Variables

This program will find all prime factors of a positive integer n, or list all prime numbers between lower and upper bounds specified by the user.

A routine under LBL a is used in determining the factors of an integer n. This routine selects a trial divisor d and tests d as a factor of n. If d divides n, then $n \leftarrow n/d$ and d is tested as a factor of the new n. If d does not divide n, a new d is selected. The process continues until $d > \sqrt{n}$, at which point n is returned as the final factor. The trial divisor d takes on the values 2,3,5,7; then for $d > 10$, d takes on those values that satisfy $(d - 10) \bmod 30 = 1, 3, 7, 9, 13, 19, 21,$ or 27. Thus in the first cycle of 30 integers from 11 to 40, d takes on the values 11, 13, 17, 19, 23, 29, 31, and 37. This technique eliminates from the test those values of d ($d>10$) which are divisible by 2, 3, or 5.

To list primes, a lower bound for the search must be specified. The upper bound is an optional input; if omitted, a default value of 2×10^9 is assumed. Upper and lower bounds need not be integers. The search for primes also uses LBL a to determine if an integer n has any factors or is indeed prime. Once an integer n ($n \geq 3$) has been tested and found to be either prime or non-prime, the next integer tested is n+2.

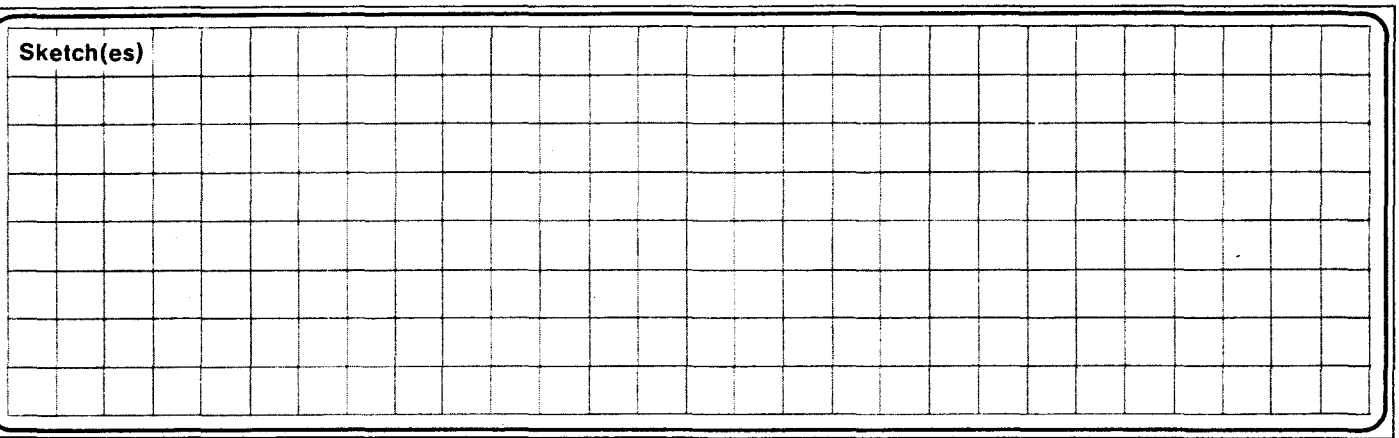
Remarks:

1. The number n to be factored must be an integer in the range $0 < n \leq 2 \times 10^9$. Any other input will result in a program halt with a display of "Error".
2. The upper bound of the search for primes must be greater than or equal to the lower bound, or an Error halt will occur.
3. AUTO mode is available to allow automatic output of all calculated results through PRINT/PAUSE commands. The end of all calculations is signalled by a 0.00 in the display for both modes.
4. Either routine can be quite time-consuming for large integers.

Operating Limits

Program Description 11

Sketch(es)



Sample Problem(s)

Example 1:

Find the prime factors of 924. Do not set AUTO mode.

Keystrokes:	Outputs:	
924 A →	2.00	
R/S →	2.00	
R/S →	3.00	
R/S →	7.00	
R/S →	11.00	
R/S →	0.00	(end)

Thus $924 = 2 \times 2 \times 3 \times 7 \times 11$.

Example 2:

Find the prime factors of 3623. Do not use AUTO mode.

Keystrokes:	Outputs:	
3623 A →	3623.00	
R/S →	0.00	(end)

3623 is prime.

Solution(s)

Example 3:

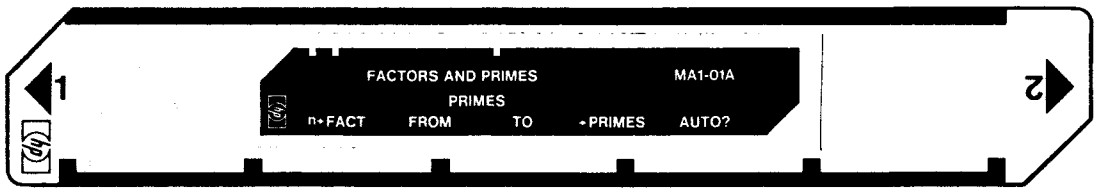
Find all prime numbers between 101 and 125. Use AUTO mode.

Keystrokes:	Outputs:	
101 B 125 C E →	1.00	(AUTO set)
D →	101.00 ***	
	103.00 ***	
	107.00 ***	
	109.00 ***	
	113.00 ***	
	0.00	(end)

Reference(s)

00063D

User Instructions



STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS	OUTPUT DATA/UNITS
1	Load side 1 and side 2.			
2	To allow automatic output of			
	results, set AUTO mode.		E	1.00
3	To cancel AUTO mode later.		E	0.00
4	To find factors, go to step 5;			
	to find primes, go to step 6.			
	FACTORS			
5	Key in the integer and find its			
	prime factors (0.00 signals end).	n	A	Factors
				0.00
	PRIMES			
6	Key in the lower bound of the			
	search for primes.	FROM	B	FROM
7	(optional) Key in the upper			
	bound of the search (if omitted,			
	TO = 2 x 10^9).	TO	C	TO
8	Find all primes between FROM			
	and TO (0.00 signals end of			
	calculation).		D	Primes
				0.00

Factors and Primes

STEP	KEY ENTRY	COMMENTS
001	*LBLA	Factor integer n.
002	STOB	
003	ENT↑	
004	INT	
005	X≠Y?	If non-integer, halt on Error.
006	GTO5	
007	0	
008	STOD	Initialize d.
009	X≷Y	
010	X≤Y?	If n ≤ 0, halt on Error.
011	GTO5	
012	2	
013	EEX	
014	9	
015	X≷Y	
016	X>Y?	If n > 2 x 10^9, halt on Error;
017	GTO5	F1 clear for factors.
018	CF1	Find factors.
019	GSBa	
020	RTN	
021	*LBLB	Lower bound for primes.
022	STOA	
023	X<0?	If negative, halt on Error.
024	GTO5	
025	ENT↑	This routine finds smallest
026	INT	potential prime ≥ user's
027	X=Y?	input.
028	GTO0	
029	1	
030	+	
031	*LBL0	
032	2	Handle 2 as special case
033	X≷Y	(only even prime).
034	X=Y?	
035	GTO0	
036	2	
037	÷	
038	INT	
039	2	
040	×	
041	1	
042	+	
043	*LBL0	
044	STOB	Store beginning prime L.
045	STOC	
046	2	2 x 10^9 is default upper
047	EEX	bound.
048	9	
049	STOE	If input ≥ 2 x 10^9, halt on
050	X≠Y?	Error.
051	GTO5	Flag 1 set for primes.
052	SF1	
053	RCLA	
054	RTN	
055	*LBLC	Upper bound for primes.
056	STOA	
057	RCLB	
058	-	
059	X<0?	If upper < lower, halt on
060	GTO5	Error.
061	RCLA	
062	INT	
063	2	Handle 2 as special case.
064	X≷Y	
065	X=Y?	
066	GTO0	
067	2	
068	÷	
069	.	This routine finds greatest
070	5	potential prime ≤ user's
071	+	input.
072	INT	
073	2	
074	×	
075	1	
076	-	
077	*LBL0	
078	STOE	
079	RCLA	Store final prime U.
080	RTN	
081	*LBLD	Routine to list primes.
082	0	
083	STOD	Initialize d←0
084	RCLB	
085	2	If L = 2, print 2, add 1
086	X≷Y	and go.
087	X=Y?	
088	GTO0	
089	1	If L = 1, print 1, 2, add 1
090	X≠Y?	and go.
091	GTO1	
092	GSBe	If L ≠ 1 and L ≠ 2, go
093	+	directly to LBL 1.
094	RCLE	
095	X≷Y	
096	X>Y?	
097	GTO4	
098	*LBL0	
099	GSBe	Output 2.
100	1	
101	STOB	
102	*LBL1	
103	GSBa	Begin main loop.
104	RCLB	Check for factors of
105	RCLC	current n (R_B).
106	X=Y?	If $R_B = R_C$, n is prime.
107	GSBe	Output n.
108	2	
109	*LBL8	
110	+	
111	STOC	Set n to next potential
112	STOB	prime.

REGISTERS

0	1	2	3	4	5	6	7	8	9
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I
Used	n	Potential prime	d	U	

Program Listing II

STEP	KEY ENTRY	COMMENTS
113	RCLE	
114	X≠Y	
115	X>Y?	
116	GTO4	If n > U, exit.
117	0	
118	STOD	
119	GTO1	Else loop again.
120	*LBLa	----------
121	2	Subroutine called from both A & D which finds factors of n.
122	GSB3	
123	X=0?	
124	RTN	
125	1	
126	GSB3	Check first if n divisible by 2, 3, 5, or 7.
127	X=0?	
128	RTN	
129	2	
130	GSB3	
131	X=0?	LBL 2 check for division by integers whose position in a cycle of 30 corresponds to 11, 13, 17, 19, 23, 29, 31, or 37.
132	RTN	
133	2	
134	GSB3	
135	X=0?	
136	RTN	
137	*LBL2	---------
138	4	
139	GSB3	11 (=7 +4)
140	X=0?	
141	RTN	
142	2	
143	GSB3	13 (=11 +2)
144	X=0?	
145	RTN	
146	4	
147	GSB3	17 (=13 +4)
148	X=0?	
149	RTN	
150	2	
151	GSB3	19 (=17 +2)
152	X=0?	
153	RTN	
154	4	
155	GSB3	23 (=19 +4)
156	X=0?	
157	RTN	
158	6	
159	GSB3	29 (=23 +6)
160	X=0?	
161	RTN	
162	2	31 (=29 +2)
163	GSB3	
164	X=0?	
165	RTN	
166	6	
167	GSB3	37 (=31 +6)
168	X=0?	

STEP	KEY ENTRY	COMMENTS
169	RTN	Loop again for next 30.
170	GTO2	----------
171	*LBL3	
172	RCLD	Tests if d|n.
173	+	
174	STOD	d ← d + x
175	RCLB	n
176	X≠Y	
177	÷	n/d
178	LSTX	d, n/d
179	X>Y?	If d > n/d, then d > $\sqrt{n}$
180	GTO0	Exit.
181	X≠Y	
182	INT	[n/d] ::= INT (n/d)
183	LSTX	n/d, [n/d]
184	X≠Y?	If non-integer, d does not divide n.
185	RTN	Else n ← n/d
186	STOE	
187	F1?	
188	CLX	If finding primes, exit.
189	F1?	
190	RTN	
191	RCLD	If factoring, output d.
192	GSBe	
193	0	Check for d a multiple factor.
194	GTO3	--------
195	*LBL0	
196	F1?	Coming here means n prime.
197	CLX	If finding primes, exit.
198	F1?	
199	RTN	Else output last n.
200	RCLB	--------
201	GSBe	
202	*LBL4	
203	CLX	Exit displaying 0.
204	F0?	
205	SPC	
206	RTN	
207	*LBLE	----------
208	F0?	Auto toggle.
209	GTO0	
210	SF0	
211	1	
212	RTN	
213	*LBLe	
214	CF0	
215	0	
216	RTN	----------
217	*LBLe	Output routine.
218	F0?	Print if AUTO.
219	PRTX	
220	F0?	
221	RTN	
222	R/S	
223	RTN	Halt if not.

LABELS					FLAGS	SET STATUS		
A n → Factors	B Primes from	C Primes to	D → Primes	E AUTO?	0 Auto	FLAGS: ON OFF	TRIG	DISP
a Factor n	b	c	d	e Output	1 Primes	0 ☐ ☑	DEG ☒	FIX ☒
0 Used	1 Primes loop	2 Divisor loop	3 d|n?	4 Exit	2	1 ☐ ☑	GRAD ☐	SCI ☐
5 Non-existent	6	7	8 L = 1 or 2	9	3	2 ☐ ☑	RAD ☐	ENG ☐
						3 ☐ ☑		n 2