

HP-41C FI Improved Version by Anthony G. Hutchins  
PPC Calculator Journal Vol 10 N1 P28 Jan-Feb 1983

```

01 LBL "FIN" ;429 BYTES, SIZE=10, FL 5-10
02 GTO IND 06 ;For internal program entry
03 LBL 11 ;Toggle INT or PMT mode and
04 FS?C IND Y ;Display Status of toggles
05 GTO 10 ; FLAGS -----
06 FC?C IND X ;6 7 8 9 | DISPLAY |
07 SF IND X ; -----
08 FC? IND X ;* - - * INT:B, PMT:B
09 SF IND Y ;* * - - INT:B, PMT:C
10 LBL 10 ;* - - - INT:B, PMT:E
11 8
12 6 ;- - * * INT:C, PMT:B
13 " INT:" ;- * * - INT:C, PMT:C
14 XEQ 13 ;- - * - INT:C, PMT:E
15 7
16 9 ;- - - * INT:E, PMT:B
17 ">" PMT:" ;- * - - INT:E, PMT:C
18 XEQ 13 ;- - - - INT:E, PMT:E
19 AVIEW
20 LBL 12 ;Position pointer at top
21 CF 22 ; Function Key Directory.
22 RTN
23 LBL b ;Access to INT toggle routine
24 8
25 6 ;Initialize for flags 6 & 8.
26 GTO 11
27 LBL c ;Access to PMT toggle routine
28 7
29 9 ;Initialize for flags 7 & 9.
30 GTO 11
31 LBL e ;Clear Financial Registers
32 LBL 00 ; and set default values.
33 SREG 01 ;SigmaREG
34 CLS ;CLSigma
35 E ;Set R 01 thru R 05 = 0
36 STO 08 ;R 08 = 1 (CF)
37 STO 09 ;R 09 = 1 (PF)
38 LBL J ;Access to Status Display
39 GTO 10
40 LBL H ;Enter: CF
41 STO 08 ; (Compounding Frequency)
42 GTO 12
43 LBL I ;Enter: PF
44 STO 09 ; (Payment Frequency)
45 GTO 12
46 LBL a ;Multiply contents of I reg
47 RCL 09 ; by PF and store as n.
48 *
49 LBL A ;Enter or Solve for: n
50 E ; (Number of Periods)
51 GTO 00
52 LBL B ;Enter or Solve for: i%
53 2 ; (Interest Rate)
54 GTO 00
55 LBL C ;Enter or Solve for: PV
56 3 ; (Present Value)
57 GTO 00
58 LBL d ;Divide contents of I reg
59 RCL 09 ;by PF and store as PMT
60 /
61 LBL D ;Enter or Solve for: PMT
62 4 ; (Payment)
63 GTO 00
64 LBL E ;Enter or Solve for: FV
65 5 ; (Future Value)
66 LBL 00 ;Data entry routine stores
67 STO 06 ;all inputs and solutions

```

```

68 X<>Y      ;in the Financial Registers
69 FC? 22    ;and positions the program
70 XEQ IND 06 ;pointer at the top of the
71 STO IND 06 ;Function Key Directory.
72 GTO 12
73 LBL 01     ;Solve for: n  (%i = 0)
74 RCL 02
75 X#0?
76 GTO 01     ;      -[PV+FV]
77 RCL 04     ;  n = -----
78 RCL 03     ;      PMT
79 GTO 00
80 LBL 01     ;Solve for: n  (%i <> 0)
81 XEQ 06
82 RCL 03     ;      PV+FV
83 +         ;  LN[ 1 - ----- ]
84 LASTX      ;      PV+C
85 XEQ 00     ;  n = -----
86 LN1+X      ;      j
87 RCL 07
88 /
89 RTN
90 LBL 02     ;Solve for: %i
91 RCL 04
92 X#0?
93 GTO 02     ;<-- exit if PMT <> 0
94 RCL 05
95 RCL 03     ;      for PMT = 0
96 /
97 CHS       ;      LN[-FV/PV]
98 LN        ;  j = -----
99 RCL 01     ;      n
100 /
101 STO 07
102 GTO 09     ;<-- exit to convert j to %i
103 LBL 03     ;Solve for: PV
104 XEQ 06
105 *         ; %i <> 0:  PV = -[BC+FV]/A
106 GTO 00     ; %i = 0:  PV = -[nPMT+FV]
107 LBL 04     ;Solve for: PMT
108 XEQ 06     ;      -b[PV*A+FV]
109 X<> T      ; %i <> 0:  PMT = -----
110 /         ;      B
111 X<>Y      ;      -[PV+FV]
112 RCL 03     ; %i = 0:  PMT = -----
113 *         ;      n
114 LBL 00     ;Common subroutine
115 RCL 05     ;      X+FV
116 +         ;  X = - [-----]
117 X<>Y      ;      Y
118 /         ;  Executed when solving
119 CHS       ;  for: n, PV and PMT.
120 RTN
121 LBL 05     ;Solve for: FV
122 XEQ 06
123 *         ; %i <> 0:  FV = -[PV*A+BC]
124 X<>Y
125 RCL 03     ; %i = 0:  FV = -[PV+nPMT]
126 *
127 +
128 CHS
129 RTN
130 LBL 06     ;Convert: %i --> i --> j
131 E         ;  (inverse of LBL 09)
132 RCL 02
133 %         ;  i = %i / 100
134 FS? 08
135 GTO 00     ; for INT mode: B
136 RCL 08     ;  j = -CF[LN(1-i/CF)]/PF
137 FS? 06

```

```

138 CHS      ; for INT mode: C
139 /        ; j = i/PF
140 LASTX
141 X<>Y      ; for INT mode: E
142 LN1+X     ; j = CF[LN(1+i/CF)]/PF
143 *
144 LBL 00
145 RCL 09
146 /
147 STO 07    ;<-- store j
148 X#0?
149 GTO 07    ;<-- exit if j <> 0
150 SIGN      ;else: for j = 0   T | 1
151 RCL 01    ; load stack for   Z | 1
152 RCL 04    ; %i = 0 case.     Y | n
153 RTN       ;                  X | PMT
154 LBL 07    ;Develop compound interest
155 RCL 04    ; factors in stack.
156 RCL 07    ;[common to all solutions]
157 FS? 09
158 CHS      ;FUNDAMENTAL EQUATION:
159 FC? 07    ; (relating all variables)
160 E^X-1     ; PV*A + B*C + FV = 0
161 FS? 09    ;where:
162 CHS      ;for PMT mode: B b = 1 - e^-j
163 /
164 LASTX     ; Stack      : C b = j
165 RCL 01    ;-----
166 RCL 07    ; T | b      : E b = e^j - 1
167 *         ; Z | A      A = e^nj
168 E^X       ; Y | B      B = e^nj - 1
169 LASTX     ; Z | C      C = PMT/b
170 E^X-1     ;-----+-----
171 R^        ; L | nj     (nj - not used)
172 RTN
173 LBL 02    ;Solve for: %i (PMT <> 0)
174 RCL 03
175 FS? 09
176 +
177 RCL 04    ;If initial cashflow & PMT
178 *         ;are both non zero and have
179 CF 05     ;opposite signs: = PV case.
180 X<0?
181 SF 05     ;SF 05 to identify PV case.
182 LASTX     ;Check for: %i = 0
183 RCL 05
184 RCL 03    ;Compute: X = (PV+FV)/n
185 +
186 RCL 01    ; if: X + PMT = 0
187 /
188 +         ; then: %i = 0
189 X=0?
190 RTN       ;<-- exit if %i = 0
191 LASTX     ;Solve for Interest rate %i
192 RCL 05    ; by computing "force of
193 RCL 03    ; interest" factor per
194 -         ; PMT period (j) which
195 X<>Y      ; is converted to %i.
196 FS? 07    ; j(0) = Initial Guess
197 CLX       ; PMT + X
198 FC? 09    ; j(0) = 2[-----]
199 CHS      ; (FV-PV) + (k*X)
200 +         ;where:
201 /         ; k = +1 for PMT mode: B
202 ST+ X     ; k = 0 for PMT mode: C
203 STO 07    ; k = -1 for PMT mode: E
204 2 E2
205 RCL 01    ; 200
206 /         ; Limit: j(0) to ---
207 X<Y?     ; n

```

```

208 STO 07
209 XEQ 02      ;Compute: j(1)
210 2 E2
211 RCL 01      ;          200
212 /          ; Limit: j(1) to ---
213 X<Y?        ;          n
214 STO 07
215 LBL 08      ;Iterative loop:
216 XEQ 02      ; computes successive
217 LASTX       ; approximations of j
218 %CH         ; until %CH = 0 after
219 RND         ; rounding.
220 X#0?        ;If not = 0 then
221 GTO 08      ;<-- repeat loop
222 LBL 09      ;Convert:    j --> i --> %i
223 CLD         ; to complete %i solution.
224 RCL 07      ; (inverse of LBL 06)
225 RCL 09
226 *
227 FS? 08      ;For INT mode: B
228 GTO 00      ;    i = CF[1-e^(-jPF/CF)]
229 RCL 08
230 FS? 06      ;for INT mode: C
231 CHS         ;    i = jPF
232 /
233 LASTX       ;for INT mode: E
234 X<>Y        ;    i = CF[e^(jPF/CF) - 1]
235 E^X-1
236 *
237 LBL 00
238 E2
239 *          ;and: %i = 100i
240 RTN
241 LBL 02      ;Subroutine: Compute
242 XEQ 07      ; next approximation of j.
243 *
244 LASTX       ;Initialize stack:
245 RCL 03
246 R^         ;      T |      BC
247 *          ;      Z |      C
248 RCL 05      ;      Y |     PV*A
249 FS? 05      ;      X |     FV
250 GTO 00
251 X<> T       ;Revise stack:  FV Case
252 +
253 ST+ Y       ;      T |      BC
254 LASTX       ;      z |    (PV+C)*A
255 X<> T       ;      Y |    (PV*A)+BC
256 GTO 02      ;      X |     FV
257 LBL 00
258 ST- Z       ;Revise stack:  PV Case
259 R^
260 +          ;      T |      BC
261 X<>Y        ;      z |     C-FV
262 LASTX       ;      Y |    BC+FV
263 RDN         ;      X |    PV*A
264 LBL 02      ;Compute next approximation
265 X<>Y        ;of j      using Newtonian
266 /          ;iteration to solve f(j)=0
267 LASTX       ;where f(j)=LN(X/Y)
268 X<>Y
269 CHS
270 LN
271 *
272 X<>Y
273 RCL 01
274 *
275 R^
276 RCL 07
277 FS? 10      ;<-- if flag 10 set, view

```

```

278 VIEW X      ;each approximation of: j
279 FC? 09      ;Compute:
280 CHS
281 FC? 07      ;for PMT mode:B b* = e^j - 1
282 E^X-1       ;           :C b* = j
283 FC? 09      ;           :E b* = 1-e^(-j)
284 CHS
285 /           ;           Y*LN(-X/Y)
286 -           ;j(n+1) = j(n) + -----
287 /           ;           (n*Z)-(T/b*)
288 RCL 07
289 +
290 STO 07      ;<-- store j(n+1)
291 RTN
292 LBL 13      ;Append: "B", "C" or "E"
293 FS? IND Y   ;to status display based on
294 >"C"        ;status of flags specified
295 FS? IND Y   ;in x and Y registers.
296 RTN
297 FC? IND X   ;If flag Y is set: mode = C
298 >"E"        ;else
299 FS? IND X   ;If flag X is set: mode = B
300 >"B"        ;else:           mode = E
301 END

```

```

01ADC000F40046494EAE060CAAF2BB00ABF3A8F3ADF3A8F20B
180016F520494E543AE0000D170019F67F20504D543AE0000D
7E0DA91685CF7C180016BC00CF7D170019BC00CF7F01990170
1B3839CF6FBB00CF6D38BD00CF6E39BD00CF7B2942CF661BB1
00CF6712B100CF6813B100CF7E2943CF6914B100CF6A150136
71AD16AE869186BD00022263B2002423B10002E00006234076
E0000065274385032463B3002523435450214337BA0004E000
0642B10005E00006CE70437123420125407143548506E00006
42712342405485071B224CAC08B10028AC0654437671654201
29433763B8007A212485082427AC0954AD0758AC0954437621
274255765874850323AC09402442A90566A805762523402143
4067857625234171AC0777AD09544043927337121B12214344
37E00002121B122143443709E00002764D6E63B9000A7F2729
42AC08B10028AC06544376715842011B12428503E000074276
23744225AC05B100CE7040927276CE70B30001937174407176
7503714376715450427121427427AC0A9873AD0954AD0758AD
0954434143274037850EACF2F27F43ACF285ADF3F27F45ACF3
F27F42C0000D28

```

429 BYTES

# HP-41C FI Improved by Anthony G. Hutchins PPC V10 N1 P28 Jan-Feb 1983

Program Registers Needed: 62

Row 1 (1 - 5)



Row 2 (5 - 12)



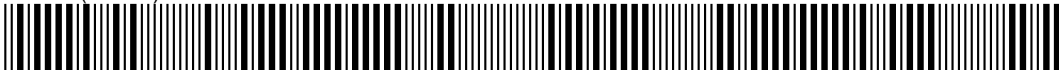
Row 3 (13 - 17)



Row 4 (17 - 21)



Row 5 (22 - 29)



Row 6 (30 - 38)



Row 7 (39 - 46)



Row 8 (46 - 54)



Row 9 (55 - 63)



Row 10 (63 - 71)



Row 11 (72 - 81)



Row 12 (81 - 90)



Row 13 (91 - 102)



Row 14 (102 - 109)



Row 15 (109 - 121)



Row 16 (122 - 132)



# HP-41C FI Improved by Anthony G. Hutchins PPC V10 N1 P28 Jan-Feb 1983

Row 17 (133 - 142)



Row 18 (143 - 154)



Row 19 (155 - 164)



Row 20 (165 - 176)



Row 21 (177 - 187)



Row 22 (188 - 198)



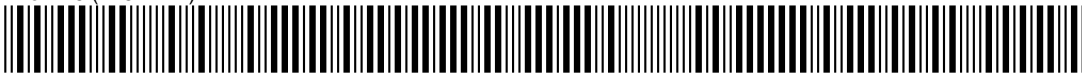
Row 23 (199 - 208)



Row 24 (209 - 216)



Row 25 (216 - 227)



Row 26 (227 - 237)



Row 27 (238 - 247)



Row 28 (248 - 255)



Row 29 (256 - 266)



Row 30 (267 - 278)



Row 31 (278 - 287)



Row 32 (288 - 296)



Row 33 (297 - 301)

