INPUT N
=
OF DIGITS

```
0000 `FRUIT PI+` PAU PAU
0011 CE CHR 035 ` OF DI`
0021 `GITS?` BRK STO 000
```

N MUST BE NO LARGER THAN 2070

```
0030 2070 INV IF< 000
0038 GTO 0059 `TOO MANY`
0049 ` DIGITS` GTO 0009
```

9 OR 10 PER REGISTER

```
0059 CFG 770 INV IF< 000
0067 SF 03 800.2600 PAR
```

INITIALIZATION

```
0078 CLR x~t RCL 000 CMS
0084 FIX 9 STO 474 +8+0
0093 TF 03 1=/(9+0 TF 03
0104 1)= INT STO 473
```

PART A

16 ARCTAN (1/5)

```
0111 RCL 474 +1=/5 LOG +
0121 1=/2= INT STO 000 1
0131 STO 003 (9+0 TF 03
0140 1) INV LOG STO 004
0147 *80= STO 238 238
0157 STO 001 8 STO 002
0164 RCL 473 STO 005 CLR
0171 STO 006 STO 471
0177 STO 472 RCL IND 001
0184 INV IF= 472
0188 GTO 0204 INC 001
0194 INC 002 INV INC 005
0201 GTO 0180 RCL 471 *
0208 RCL 004 +
0212 RCL IND 001 -( CE /
0220 25) INT STO IND 001
0228 *25= STO 471
0235 RCL 006 * RCL 004 +
0243 RCL IND 001 -( CE /
0251 RCL 003 ) INT
0256 STO 006 * RCL 003 =
0264 EXC 006 TF 01 +/-
0270 ST+ IND 002 INC 001
0277 INC 002 DSZ 005
0283 GTO 0204 TF 01
0288 GTO 0296 SF 01
0293 GTO 0298 RF 01
0298 INC 003 INC 003
0304 DSZ 000 GTO 0154
```

PART B

4ARCTAN($\frac{1}{239}$)

```
0310 RCL 474 +1=/239 LOG
0321 +1=/2= INT STO 000
0331 1 STO 003 956*
0339 RCL 004 = STO 238
0346 SF 01 238 STO 001 8
0355 STO 002 RCL 473
0361 STO 005 CLR STO 006
0368 STO 471 STO 470
0374 STO 472 RCL IND 001
0381 INV IF= 472
0385 GTO 0401 INC 001
0391 INC 002 INV INC 005
0398 GTO 0377 RF 02
0403 GTO 0408 SF 02
0408 RCL 471 x~t RCL 470
0415 STO 471 x~t *
0420 RCL 004 +
0424 RCL IND 001 -( CE /
0432 239) INT
0437 STO IND 001 *239=
0446 STO 470 INV TF 02
0452 GTO 0406 RCL 006 *
0459 RCL 004 +
0463 RCL IND 001 -( CE /
0471 RCL 003 ) INT
0476 STO 006 * RCL 003 =
0484 EXC 006 TF 01 +/-
0490 ST+ IND 002 INC 001
0497 INC 002 DSZ 005
0503 GTO 0401 TF 01
0508 GTO 0516 SF 01
0513 GTO 0518 RF 01
0518 INC 003 INC 003
0524 DSZ 000 GTO 0348.
```

PART C

```
0530 RCL 473 STO 001 +7=
0539 STO 002 RCL IND 002
0546 EXC 472 RCL 004 INV
0553 IF< 472 GTO 0577
0559 ST- IND 002 INV
0564 INC 002 INC IND 002
0571 INC 002 GTO 0542
0577 CLR STO 472
0581 RCL IND 002 INV
0586 IF< 472 GTO 0614
0592 RCL 004 ST+ IND 002
0599 INV INC 002 INV
0604 INC IND 002 INC 002
0611 GTO 0577 INV
0615 INC 002 DSZ 001
0621 GTO 0542 INV TF 03
```

DIRECT PRINTING OF 10 DIGIT REGISTERS

```
0627 GTO 0663 7 STO 001
0634 RCL 473 +1= STO 002
0643 RCL IND 001 PRT INV
0649 TF 74 BRK INC 001
0655 DSZ 002 GTO 0643
0661 CLR HLT RCL 007 PRT
```

CHANGE 9 DIGIT REGISTERS TO 10 DIGIT GROUPS AND PRINT

```
0667 INV TF 74 BRK 0
0672 STO 000 8 STO 001 9
0680 STO 002 (
0684 RCL IND 001 IF= 000
0691 HLT INC 001 /
0696 RCL 002 INV INC 002
0703 INV LOG ) FRC *10
0710 INV LOG +(
0714 RCL IND 001 /
0719 RCL 002 INV LOG )
0725 INT = PRT INV TF 74
0731 BRK CLR INV IF= 002
0737 GTO 0683 INC 001 9
0744 ST+ 002 GTO 0683
```

LOTS OF π   by Bob Fruit

The "Calcu-Letter" column of Popular Science (July, 1981) reported that the Hewlett-Packard calculator club had calculated the value of π to 1,000 decimal places in less then 11½ hours and had challenged the Texas Instruments Personal Calculator Club to try and beat their time. This got my interest on the subject of calculating the value of π on my TI-59. I had never considered trying to calculate the value of π, particularly to 1,000 places.

I started by writing to both the TI club (to join it) and the H-P club (to get a copy of the article on their calculating the value of π). Not waiting for the replies I went about figuring out how I would tackle this problem. That effort led to program I. It is shown here because it makes it easier to follow the programming techniques and algorithms developed here, but also used in my later programs. (There are many improvements that are obviously needed in program I, but I do not want to dwell on them here. Remember it is just to help you see what's going on in the later program.)

When considering doing a problem like π you must determine how to do two things: First, what algorithm to use for calculating the value of π? I looked in Petr Beckman's book A History of π (this book is much more entertaining than the title might imply) to see what algorithms had been used in the past. The one I liked was;

$$\pi = 16 * \arctan(1/5) - 4 * \arctan(1/239).$$

This turned out to be the same algorithm the H-P people used. The arctan function has a nice numerical method;

$$\arctan(X) = X - X^3/3 + X^5/5 - X^7/7 + ...$$

The second thing needed is, how to do the multi-precision arithmatic. The multi-precision addition and subtraction is not to hard to figure out, but the multi-precision division looked like a formidable challenge. It turns out that a multi-precision numerator divided by a single precision denominator (the situation faced in this problem) is easy. Look at each block of digits of the multi-precision numerator as a stand alone single precision number. Carry out the normal division between two numbers, save the dividend (this is part of the multi-precision answer), and add the remainder onto the front of the next block of digits. Repeat the division process untill the entire multi-precision division is complete. This is just the way long division is taught in grade school, it just looks a little different.

Since the largest remainder could be 238 not more then 10 digits can be stored in each register or you would exceed the 13 digit accuracy of the TI-59 (and there are some

obvious printing advantages too). This means that I will only get 460 decimal places in my calculation of π. I would need 207 registers to be able to get 1,000 decimal places. As best I can determine (I can not really read H-P program code) the H-P people used 205 registers in their effort on π. Since the H-P calculator has more user memory they will be able to calculate more decimal places using similar algorithms.

The procedure for running programs II are: Load all the programs on to magnetic cards (the calculator configurations are II-A 479.59, II-B 159.99, II-C 159.99). Make sure that registers 90-99 are all zero when saving program II-B. Read in program II-A and press A. Read in program II-A. Put program II-B in the card reader (it will be read when the calculator is ready for it). After program II-B has been read put program II-C into the card reader. To print additional copies of the answer use INV LIST. The last digit is too large by 4 (the last 3 digits should be 799).

What happens while the program is running? Program II-A sets the calculator in fast mode and calculates the value of 16 * arctan(1/5). Program II-B calculates the value of 4 * arctan(1/239) and subtracts it from the value found in the first program. Program II-C takes the value of π and performs all the carry/borrow 1 operations that were ignored during the addition/subtractions in the first two programs and prints the results.

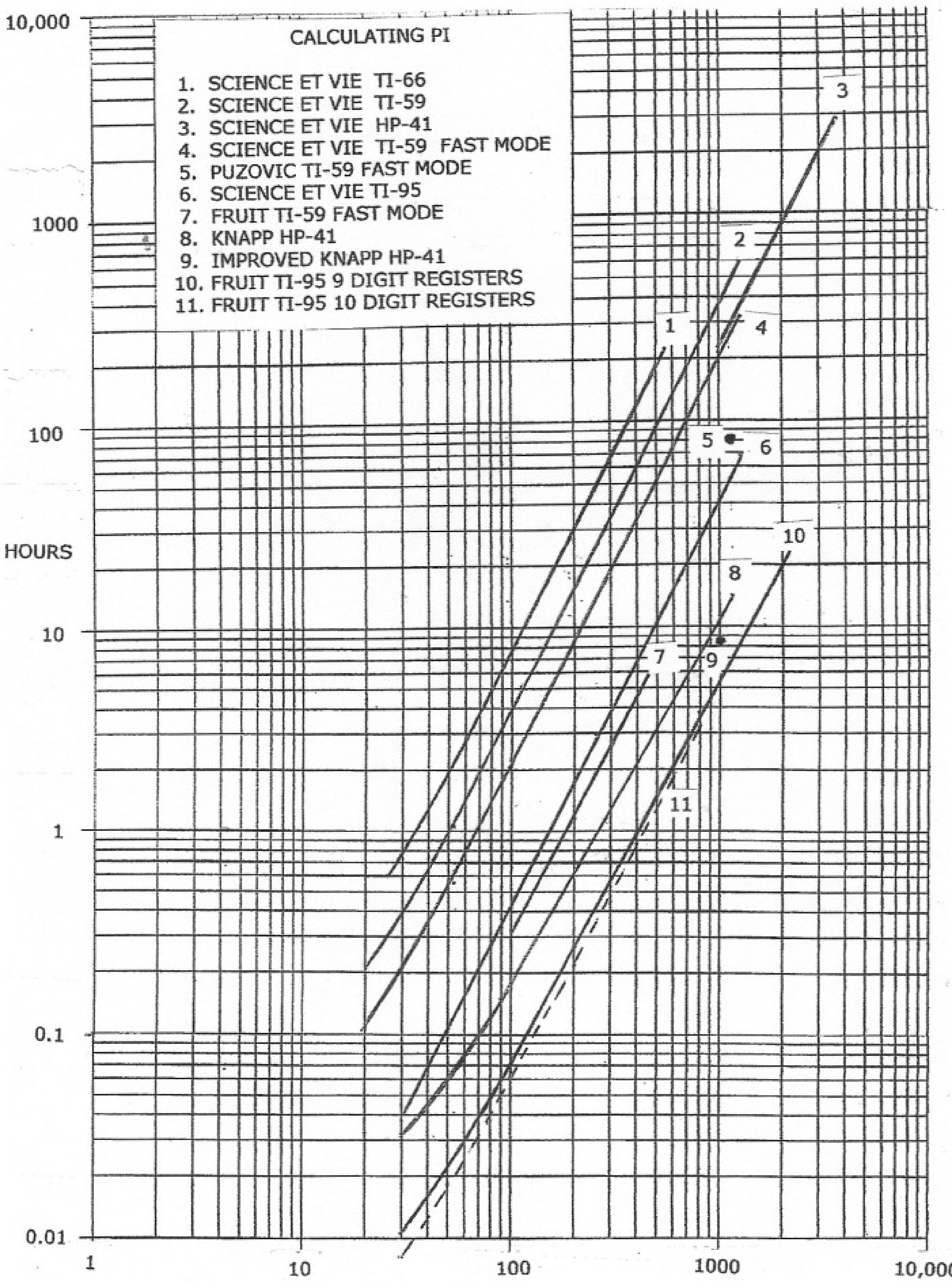
Programs II will calculate the value of π to 460 decimal places (with the error noted before) in 6 hours 18 minutes 15 seconds. For program II-A the running time is 4:14:51. Program II-B's running time is 2:03:24. I do not count the couple of minutes of program II-C running time since that is not part of the calculation of π.

Now for the diappointing part. It is easy to project the running time if 1,000 decimal places could be calculated on a TI-59. There is a linear relationship between the number of decimal places and the running time of the prorams, so 1000 / 460 * (6:18:15) = 13:46. This puts my approach 2¼ hours longer than the H-P effort.

Time will not permit me to get back into this problem now. If I do come back to this problem I will look at incororating the algabric contraction the H-P people developed for the formulas. I don't know if that would significantly improve the run times I have already gotten. I would like to thank Rich Nelson, editor for the H-P newsletter, for sending me a copy of their article π. I had sent him a SASE with my request.

LOTS OF PIE, by Bob Fruit.(no pun intended) Bob has risen to the challenge and
----------- the problem of PI. I hope that this first "sheep" being over the
will be followed by the whole herd. Let's show the "friends" from the HP PPC C
we are not beaten yet, even we lost (temporarily) the battle of the calendar,
intend to loose the whole "war."

On the next page you will find a table of PI computed to the first 1
It was obtained in July 1961 by Shanks and Wrench on an IBM 7090. At that tim
computed PI to 100,265 places. This table was reprinted with permission of th
The American Mathematical Society, from Mathematics of Computation, 1962,v14n



CALCULATING PI
1. SCIENCE ET VIE TI-66
2. SCIENCE ET VIE TI-59
3. SCIENCE ET VIE HP-41
4. SCIENCE ET VIE TI-59 FAST MODE
5. PUZOVIC TI-59 FAST MODE
6. SCIENCE ET VIE TI-95
7. FRUIT TI-59 FAST MODE
8. KNAPP HP-41
9. IMPROVED KNAPP HP-41
10. FRUIT TI-95 9 DIGIT REGISTERS
11. FRUIT TI-95 10 DIGIT REGISTERS
HOURS
10,000
1000
100
10
1
0.1
0.01
1
10
100
1000
10,000
1
2
3
4
5
6
7
8
9
10
11