

```

5310 11720 LXI H,DOCMP ;NUMERIC CASE
5319 11740 PUSH H ;CODE BACK TO COMPARE
5320 11760 JZ FCOMP
5321 11780 JZ FCOMP ;DO NUMERIC COMPARE
5322 11800 XRA A ;SET VALUE TYPE AS NUMERIC
5323 11820 STA VALTYP>>>
5324 11840 IFN STRING,<
5325 11860 ;
5326 11880 ; THE FOLLOWING ROUTINE COMPARES TWO STRINGS
5327 11900 ; ONE WITH DESC IN [D,E] OTHER WITH DESC, IN [FACLO, FACLO+1]
5328 11920 ; A=0 IF STRINGS EQUAL
5329 11940 ; A=377 IF B,C,D,E POINTER FACLO
5330 11960 ; A=1 IF B,C,D,E ,LT, FACLO
5331 11980 ;
5332 12000 IFN LENGTH=2,<
5333 12020 STRCMP: PUSH D ;SAVE DESC, POINTER TO FIRST STR,
5334 12040 CALL PREFAC ;FREE THE FACLO STR
5335 12060 POP D ;RESTORE 1ST DESC, POINTER
5336 12080 PUSHM ;SAVE LENGTH
5337 12100 PUSHM ;SAVE POINTER
5338 12120 CALL FRETMP ;FREES 1ST DESC, POINTER
5339 12140 CALL MOVHM ;[B,C] POINT AT FIRST CHAR
5340 12160 ;[E] HAS THE LENGTH
5341 12180 POP H ;GET 2ND CHARACTER POINTER IN H
5342 12200 XTML MOV D,L> ;GET 2ND CHARACTER COUNTER IN L
5343 12220 ;
5344 12240 IFE LENGTH=2,< ;SAVE IN D
5345 006320* 001000 000315 12260 STRCMP: CALL FRESTH ;FREE UP THE FAC STRING, AND GET THE
5346 006321* 000000 010431*
5347 006322* 000000 006310*
5348 ;
5349 12280 ;POINTER TO THE FAC DESCRIPTOR IN [M,L]
5350 006323* 001000 000176 12300 MOV A,M ;SAVE THE LENGTH OF THE FAC STRING IN [A]
5351 006324* 001000 000043 12320 INX H
5352 006325* 001000 000116 12340 MOV C,M ;SAVE THE POINTER AT THE FAC STRING
5353 006326* 001000 000043 12360 ;DATA IN [B,C]
5354 006327* 001000 000106 12380 INX H
5355 006330* 001000 000321 12400 MOV B,M ;
5356 006331* 001000 000305 12420 POP D ;GET THE STACK STRING POINTER
5357 006332* 001000 000365 12440 PUSH PSW ;SAVE THE POINTER AT THE FAC STRING DATA
5358 006333* 001000 000315 12460 CALL FRETMP ;SAVE THE FAC STRING LENGTH
5359 006334* 000000 010440* ;FREE UP THE STACK STRING AND RETURN
5360 006335* 000000 006321*
5361 ;
5362 12480 ;THE POINTER TO THE STACK STRING DESCRIPTOR
5363 006336* 001000 000321 12500 JIN [M,L]
5364 006337* 001000 000136 12520 POP D ;[D]=LENGTH OF FAC STRING
5365 006340* 001000 000043 12540 MOV E,M ;[E]=LENGTH OF STACK STRING
5366 006341* 001000 000116 12560 INX H
5367 006342* 001000 000043 12580 MOV C,M ;[B,C]=POINTER AT STACK STRING
5368 006343* 001000 000106 12600 INX H
5369 006344* 001000 000341 12620 MOV B,M> ;
5370 006345* 001000 000173 12640 POP H ;GET BACK 2ND CHARACTER POINTER
5371 ;CSLOOP: MOV A,E ;BOTH STRINGS ENDED

```

```

5371 12660 ORA D ;TEST BY OR'ING THE LENGTHS TOGETHER
5372 12700 RZ ;IF SO, RETURN WITH A ZERO
5373 12720 MOV A,D ;GET FACLO STRING LENGTH
5374 006351* 001000 000267 12740 ORA A ;IF IT ENDED, OTHER MUST NOT HAVE
5375 006352* 001000 000057 12760 CMA ;
5376 006353* 001000 000310 12780 RZ ;TEST
5377 006354* 001000 000257 12800 XRA A ;MUST NOT HAVE BEEN ZERO, TEST CASE
5378 006355* 001000 000273 12820 CMP E ;OF B,C,D,E STRING HAVING ENDED FIRST
5379 006356* 001000 000074 12840 INR A ;RETURN WITH A=1
5380 006357* 001000 000320 12860 RNC ;TEST THE CONDITION
5381 ;
5382 006360* 001000 000025 12880 JHERE WHEN NEITHER STRING ENDED
5383 006361* 001000 000035 12900 DCR D ;DECREMENT BOTH CHARACTER COUNTS
5384 006362* 001000 000012 12920 DCR E
5385 006363* 001000 000276 12940 LOAX B ;GET CHARACTER FROM B,C,D,E STRING
5386 006364* 001000 000043 12960 CMP M ;COMPARE WITH FACLO STRING
5387 006365* 001000 000003 12980 INX H ;BUMP POINTERS (INX DOESNT CLOBBER CC'S)
5388 006366* 001000 000312 13000 INR B
5389 006367* 000000 006345* 13020 JZ CSLOOP ;IF BOTH THE SAME, MUST BE MORE TO STRINGS
5390 006370* 000000 006334*
5391 006371* 001000 000077 13040 CMC ;
5392 006372* 001000 000363 13060 JNC SIGNS> ;HERE WHEN STRINGS DIFFER
5393 006373* 000000 000000* ;SET [A] ACCORDING TO CARRY
5394 006374* 000000 006367*
5395 ;
5396 006375* 001000 000074 13080 IFN LENGTH,<
5397 006376* 001000 000217 13100 DOCMP: INR A ;SETUP BITS
5398 006377* 001000 000301 13120 ADC A ;IF LESS 2=EQUAL 1=GREATER
5399 006400* 001000 000240 13140 POP B ;WHAT DID HE WANT?
5400 006401* 001000 000306 13160 ANA B ;ANY BITS MATCH?
5401 006402* 000000 000377 13180 ADI 255 ;MAP 0 TO 0
5402 006403* 001000 000037 13200 SBB A ;
5403 ;AND ALL OTHERS TO 377
5404 006404* 001000 000315 13220 IFE LENGTH=2,<
5405 006405* 000000 000000* 13240 CALL CONIA#> ;CONVERT [A] TO AN INTEGER SIGNED
5406 006406* 000000 006375*
5407 006407* 001000 000303 13260 JMP RETADP> ;RETURN FROM OPERATOR APPLICATION
5408 006410* 000000 005354*
5409 006411* 000000 006405*
5410 ;
5411 13280 IFN LENGTH=2,<
5412 13300 JMP FLOAT ;MAKE FAC=[A] SIGNED
5413 13320 ;COULD FALL INTO FLOAT BUT MESSY (SAVES
5414 13340 ;TWO BYTES)
5415 13360 ;
5416 13380 NOTER: MVI D,90 ;*NOTH HAS PRECEDENCE 90
5417 13400 CALL LPOPER ;GO PERFORM
5418 13420 IFN STRING,<CALL CHKNUM> ;MAKE SURE ITS INT
5419 13440 CALL DEINT ;GET VALUE IN [D,E]
5420 13460 MOV A,E ;
5421 13480 CMA ;COMPLEMENT
5422 13500 MOV C,A ;[C] LOW ORDER OF ANSWER
5423 13520 MOV A,D ;
5424 13540 CMA ;COMPLEMENT HIGH ORDER TOO

```

FORMULA EVALUATOR

```

5424      13560      CALL      GIVACF      ;FLOUT (A,C) AS RESULT
5425      13580      POP        B           ;TAKE RETURN ADDRESS OF FRMEVL
5426      13600      JMP        RETAOP>>> ;OFF AND RETURN TO THE RIGHT
5427      13620      ;                     ;PLACE SO THE TEXT POINTER
5428      13640      ;                     ;WILL GET SET UP TO WHAT IT WAS
5429      13660      ;                     ;WHEN LPROPER RETURNED,
5430      13680      IFE        LENGTH=2,<
5431      000412' 001000 000026      13700      MVI        D,90      ;"NOT" HAS PRECEDENCE 90, SO
5432      000415' 000000 000132
5433      000414' 001000 000315      13720      CALL      LPROPER
5434      000415' 000000 005341'      ;FORMULA EVALUATION IS ENTERED WITH A DUMMY
5435      000416' 000000 000410'
5436
5437      000417' 001000 000315      13730
5438      000420' 000000 005573*      13740      CALL      FRCINT
5439      000421' 000000 000415'      ;ENTRY OF 90 ON THE STACK
5440      000422' 001000 000175      ;COERCE THE ARGUMENT TO INTEGER
5441      000425' 001000 000057      13760      MOV        A,L
5442      000424' 001000 000157      13780      CMA
5443      000425' 001000 000174      13800      MOV        L,A
5444      000426' 001000 000057      13820      MOV        A,H
5445      000427' 001000 000147      13840      CMA
5446      000430' 001000 000042      13860      MOV        H,A
5447      000431' 000000 001637'      13880      SHLD      FACLO
5448      000432' 000000 000420'      ;UPDATE THE FAC
5449      000433' 001000 000301
5450      13900      POP        B
5451      13910
5452      13915      ;FRMEVL, AFTER SEEING THE PRECEDENCE
5453      000434' 001000 000303      ;IF 90 THINKS IT IS APPLYING AN OPERATOR
5454      000435' 000000 005354'      ;SO IT HAS THE TEXT POINTER IN TEMP2 SO
5455      000436' 000000 000431'      ;RETURN TO REFETCH IT
5456
5457      14042      ;
5458      14044      ; DANDON APPLIES THE "AND" AND "OR" OPERATORS
5459      14046      ; AND SHOULD BE USED TO IMPLEMENT ALL LOGICAL OPERATORS,
5460      14048      ; WHENEVER AN OPERATOR IS APPLIED, ITS PRECEDENCE IS IN [8].
5461      14050      ; THIS FACT IS USED TO DISTINGUISH BETWEEN "AND" AND "OR".
5462      14052      ; THE RIGHT HAND ARGUMENT IS COERCED TO INTEGER, JUST AS
5463      14054      ; THE LEFT HAND ONE WAS WHEN IT WAS PUSHED ON THE STACK,
5464      14056      ;
5465      14060      DANDON: PUSH      B
5466      000440' 001000 000315      ;SAVE THE PRECEDENCE "OR"=70
5467      000441' 000000 000420*      ;COERCE RIGHT HAND ARGUMENT TO INTEGER
5468      000442' 000000 000435'
5469      000443' 001000 000361      14100      POP        PSW
5470      000444' 001000 000321      14120
5471      000445' 001000 000376      14140      POP        D
5472      000446' 000000 000100      14160      CPI        70
5473      000447' 001000 000173      14180      MOV        A,E
5474      000450' 001000 000312      14200      JZ        ORFIN
5475      000451' 000000 000463'      ;SETUP LOW IN (A)
5476      000452' 000000 000441'      ;DO "OR" IF PRECEDENCE WAS 70

```

```

5477      000453' 001000 000245      14220      ANA        L
5478      000454' 001000 000157      14240      MOV        L,A
5479      000455' 001000 000174      14260      MOV        A,H
5480      000456' 001000 000242      14280      ANA        D
5481      000457' 001000 000147      14300      MOV        H,A
5482      000460' 001000 000303      14320      JMP        MAKINT
5483      000461' 000000 000000*      ;RETURN THE INTEGER (H,L)
5484      000462' 000000 000451'
5485
5486      000463' 001000 000265      14322
5487      000464' 001000 000157      14324      OKFIN: ORA        L
5488      000465' 001000 000174      14326      MOV        L,A
5489      000466' 001000 000262      14328      MOV        A,H
5490      000467' 001000 000147      14330      ORA        D
5491      000470' 001000 000303      14332      MOV        H,A
5492      000471' 000000 000461*      14334      JMP        MAKINT>
5493      000472' 000000 000461*      ;RETURN THE INTEGER (H,L)
5494
5495      14336      ; AS THE "AND"ED RESULT
5496      14340      PAGE

```

Formula Evaluation

```

5496 SUBTTL DIMENSION & VARIABLE SEARCHING
5497 006473* 001000 000053 14360 DIMCON: DCX H ;SEE IF COMMA ENDED THIS VARIABLE
5498 006474* 001000 000327 14400 CHRGET ;
5499 006475* 001000 000350 14420 RZ ;
5500 006476* 001000 000317 14440 SYNCHK 44 ;IF TERMINATOR, GOOD BYE
5501 006477* 000000 000054 ;MUST BE COMMA
5502
5503 14442 ;
5504 14444 ; THE "DIM" CODE SETS DIMFLG AND THEN FALLS INTO THE VARIABLE
5505 14446 ; SEARCH ROUTINE. THE VARIABLE SEARCH ROUTINE LOOKS AT
5506 14448 ; DIMFLG AT THREE DIFFERENT POINTS:
5507 14450 ;
5508 14452 ; 1) IF AN ENTRY IS FOUND, DIMFLG BEING ON INDICATES
5509 14454 ; A "DOUBLY DIMENSIONED" VARIABLE
5510 14456 ; 2) WHEN A NEW ENTRY IS BEING BUILT DIMFLG'S BEING ON
5511 14458 ; INDICATES THE INDICES SHOULD BE USED FOR
5512 14460 ; THE SIZE OF EACH INDICE, OTHERWISE THE DEFAULT
5513 14462 ; OF TEN IS USED.
5514 14464 ; 3) WHEN THE BUILD ENTRY CODE FINISHES, ONLY IF DIMFLG IS
5515 14466 ; OFF WILL INDEXING BE DONE
5516 14468 ;
5517 000500* 001000 000001 14478 DIM: LXI B,DIMCON ;PLACE TO COME BACK TO
5518 000501* 000000 006473*
5519 000502* 000000 006471*
5520 000503* 001000 000305
5521 000504* 001000 000366
5522
5523 14480 PUSH B
5524 14482 XWD "01000","0366 ;"OKI" NON ZERO THING
5525 14484 ;
5526 14486 ; ROUTINE TO READ THE VARIABLE NAME AT THE CURRENT TEXT POSITION
5527 14488 ; AND PUT A POINTER TO ITS VALUE IN [0,e], [H,L] IS UPDATED
5528 14490 ; TO POINT TO THE CHARACTER AFTER THE VARIABLE NAME.
5529 14492 ; VALTYP IS SETUP. NOTE THAT EVALUATING SUBSCRIPTS IN
5530 14494 ; A VARIABLE NAME CAN CAUSE RECURSIVE CALLS TO PTRGET SO AT
5531 14496 ; THAT POINT ALL VALUES MUST BE STORED ON THE STACK,
5532 14498 ;
5533 000505* 001000 000257 14680 PTRGET: XRA A ;MAKE [A]=0
5534 000506* 000000 001546* 14700 STA DIMFLG ;FLAG IT AS SUCH
5535 000507* 000000 006501*
5536 000508* 000000 006501*
5537 000509* 001000 000106
5538 000510* 001000 000315
5539 000511* 000000 003612*
5540 000512* 000000 006507*
5541 000513* 001000 000332
5542 000514* 000000 002676*
5543 000515* 000000 006515*
5544 000516* 001000 000257
5545 000517* 001000 000117
5546
5547 14720 MOV B,M ;GET FIRST CHARACTER IN [B]
5548 14740 PTRGT2: CALL ISLET ;CHECK FOR LETTER
5549
5550 14760 JC SNERR ;MUST HAVE A LETTER
5551
5552 14780 XRA A
5553 14800 MOV C,A ;ASSUME NO SECOND CHARACTER
5554 14820 IFN LENGTH=2,<
5555 14840 IFN STRING,<
5556 14860 STA VALTYP>> ;DEFAULT IS ZERO (NUMERIC)
5557 14880 CHRGET ;GET THE FOLLOWING CHARACTER
5558 14900 IFE LENGTH,<
5559 14920 JNC NUSEC> ;ONLY NUMBERS ALLOWED
5560 14940 IFN LENGTH,<

```

```

5549 000523* 001000 000332 14960 JC ISSEC ;CARRY SET BY CHRGET IF CHARACTER IS
5550 000524* 000000 006534*
5551 000525* 000000 006516*
5552
5553 14980 CALL ISLET ;NUMERIC
5554 15000 ;SET CARRY IF NOT ALPHABETIC
5555
5556 15020 JC NUSEC> ;ALLOW ALPHABETICS
5557 000531* 001000 000332
5558 000532* 000000 006517*
5559 000533* 000000 006527*
5560 000534* 001000 000117
5561 000535* 001000 000327
5562
5563 15040 ISSEC: MOV C,A ;IT IS A NUMBER--SAVE IN C
5564 15060 EATEM: CHRGET ;LOOK AT NEXT CHARACTER
5565 15080 IFN LENGTH,<
5566 15100 JC EATEM ;SKIP NUMERICS
5567
5568 15120 CALL ISLET
5569
5570 15140 JNC EATEM> ;SKIP ALPHABETICS
5571
5572 15160 NOSEC:
5573 15180 IFE LENGTH=2,<
5574 15200 LXI D,HAVTYP ;SAVE JUMPS BY USING RETURN ADDRESS
5575 000547* 001000 000021
5576 000548* 000000 006574*
5577 000549* 000000 000345*
5578 000550* 001000 000325
5579 000551* 001000 000026
5580 000552* 000000 000010
5581 000553* 001000 000376
5582 000554* 000000 000045
5583 000555* 001000 000310
5584 000556* 000000 000020
5585 000557* 000000 000002
5586 000558* 001000 000376
5587 000559* 000000 000045
5588 000560* 001000 000310
5589 000561* 000000 000024
5590 000562* 001000 000376
5591 000563* 000000 000044
5592 000564* 001000 000310
5593 000565* 000000 000024
5594 000566* 001000 000376
5595 000567* 000000 000044
5596 000568* 001000 000310
5597 000569* 000000 000024
5598 000570* 001000 000376
5599 000571* 000000 000044
5600 000572* 001000 000310
5601 000573* 000000 000024
5602 000574* 001000 000376
5603 000575* 000000 000044
5604 000576* 001000 000310
5605 000577* 000000 000024
5606 000578* 001000 000376
5607 000579* 000000 000044
5608 000580* 001000 000310
5609 000581* 000000 000024
5610 000582* 001000 000376
5611 000583* 000000 000044
5612 000584* 001000 000310
5613 000585* 000000 000024
5614 000586* 001000 000376
5615 000587* 000000 000044
5616 000588* 001000 000310
5617 000589* 000000 000024
5618 000590* 001000 000376
5619 000591* 000000 000044
5620 000592* 001000 000310
5621 000593* 000000 000024
5622 000594* 001000 000376
5623 000595* 000000 000044
5624 000596* 001000 000310
5625 000597* 000000 000024
5626 000598* 001000 000376
5627 000599* 000000 000044
5628 000600* 001000 000310
5629 000601* 000000 000024
5630 000602* 001000 000376
5631 000603* 000000 000044
5632 000604* 001000 000310
5633 000605* 000000 000024
5634 000606* 001000 000376
5635 000607* 000000 000044
5636 000608* 001000 000310
5637 000609* 000000 000024
5638 000610* 001000 000376
5639 000611* 000000 000044
5640 000612* 001000 000310
5641 000613* 000000 000024
5642 000614* 001000 000376
5643 000615* 000000 000044
5644 000616* 001000 000310
5645 000617* 000000 000024
5646 000618* 001000 000376
5647 000619* 000000 000044
5648 000620* 001000 000310
5649 000621* 000000 000024
5650 000622* 001000 000376
5651 000623* 000000 000044
5652 000624* 001000 000310
5653 000625* 000000 000024
5654 000626* 001000 000376
5655 000627* 000000 000044
5656 000628* 001000 000310
5657 000629* 000000 000024
5658 000630* 001000 000376
5659 000631* 000000 000044
5660 000632* 001000 000310
5661 000633* 000000 000024
5662 000634* 001000 000376
5663 000635* 000000 000044
5664 000636* 001000 000310
5665 000637* 000000 000024
5666 000638* 001000 000376
5667 000639* 000000 000044
5668 000640* 001000 000310
5669 000641* 000000 000024
5670 000642* 001000 000376
5671 000643* 000000 000044
5672 000644* 001000 000310
5673 000645* 000000 000024
5674 000646* 001000 000376
5675 000647* 000000 000044
5676 000648* 001000 000310
5677 000649* 000000 000024
5678 000650* 001000 000376
5679 000651* 000000 000044
5680 000652* 001000 000310
5681 000653* 000000 000024
5682 000654* 001000 000376
5683 000655* 000000 000044
5684 000656* 001000 000310
5685 000657* 000000 000024
5686 000658* 001000 000376
5687 000659* 000000 000044
5688 000660* 001000 000310
5689 000661* 000000 000024
5690 000662* 001000 000376
5691 000663* 000000 000044
5692 000664* 001000 000310
5693 000665* 000000 000024
5694 000666* 001000 000376
5695 000667* 000000 000044
5696 000668* 001000 000310
5697 000669* 000000 000024
5698 000670* 001000 000376
5699 000671* 000000 000044
5700 000672* 001000 000310
5701 000673* 000000 000024
5702 000674* 001000 000376
5703 000675* 000000 000044
5704 000676* 001000 000310
5705 000677* 000000 000024
5706 000678* 001000 000376
5707 000679* 000000 000044
5708 000680* 001000 000310
5709 000681* 000000 000024
5710 000682* 001000 000376
5711 000683* 000000 000044
5712 000684* 001000 000310
5713 000685* 000000 000024
5714 000686* 001000 000376
5715 000687* 000000 000044
5716 000688* 001000 000310
5717 000689* 000000 000024
5718 000690* 001000 000376
5719 000691* 000000 000044
5720 000692* 001000 000310
5721 000693* 000000 000024
5722 000694* 001000 000376
5723 000695* 000000 000044
5724 000696* 001000 000310
5725 000697* 000000 000024
5726 000698* 001000 000376
5727 000699* 000000 000044
5728 000700* 001000 000310
5729 000701* 000000 000024
5730 000702* 001000 000376
5731 000703* 000000 000044
5732 000704* 001000 000310
5733 000705* 000000 000024
5734 000706* 001000 000376
5735 000707* 000000 000044
5736 000708* 001000 000310
5737 000709* 000000 000024
5738 000710* 001000 000376
5739 000711* 000000 000044
5740 000712* 001000 000310
5741 000713* 000000 000024
5742 000714* 001000 000376
5743 000715* 000000 000044
5744 000716* 001000 000310
5745 000717* 000000 000024
5746 000718* 001000 000376
5747 000719* 000000 000044
5748 000720* 001000 000310
5749 000721* 000000 000024
5750 000722* 001000 000376
5751 000723* 000000 000044
5752 000724* 001000 000310
5753 000725* 000000 000024
5754 000726* 001000 000376
5755 000727* 000000 000044
5756 000728* 001000 000310
5757 000729* 000000 000024
5758 000730* 001000 000376
5759 000731* 000000 000044
5760 000732* 001000 000310
5761 000733* 000000 000024
5762 000734* 001000 000376
5763 000735* 000000 000044
5764 000736* 001000 000310
5765 000737* 000000 000024
5766 000738* 001000 000376
5767 000739* 000000 000044
5768 000740* 001000 000310
5769 000741* 000000 000024
5770 000742* 001000 000376
5771 000743* 000000 000044
5772 000744* 001000 000310
5773 000745* 000000 000024
5774 000746* 001000 000376
5775 000747* 000000 000044
5776 000748* 001000 000310
5777 000749* 000000 000024
5778 000750* 001000 000376
5779 000751* 000000 000044
5780 000752* 001000 000310
5781 000753* 000000 000024
5782 000754* 001000 000376
5783 000755* 000000 000044
5784 000756* 001000 000310
5785 000757* 000000 000024
5786 000758* 001000 000376
5787 000759* 000000 000044
5788 000760* 001000 000310
5789 000761* 000000 000024
5790 000762* 001000 000376
5791 000763* 000000 000044
5792 000764* 001000 000310
5793 000765* 000000 000024
5794 000766* 001000 000376
5795 000767* 000000 000044
5796 000768* 001000 000310
5797 000769* 000000 000024
5798 000770* 001000 000376
5799 000771* 000000 000044
5800 000772* 001000 000310
5801 000773* 000000 000024
5802 000774* 001000 000376
5803 000775* 000000 000044
5804 000776* 001000 000310
5805 000777* 000000 000024
5806 000778* 001000 000376
5807 000779* 000000 000044
5808 000780* 001000 000310
5809 000781* 000000 000024
5810 000782* 001000 000376
5811 000783* 000000 000044
5812 000784* 001000 000310
5813 000785* 000000 000024
5814 000786* 001000 000376
5815 000787* 000000 000044
5816 000788* 001000 000310
5817 000789* 000000 000024
5818 000790* 001000 000376
5819 000791* 000000 000044
5820 000792* 001000 000310
5821 000793* 000000 000024
5822 000794* 001000 000376
5823 000795* 000000 000044
5824 000796* 001000 000310
5825 000797* 000000 000024
5826 000798* 001000 000376
5827 000799* 000000 000044
5828 000800* 001000 000310
5829 000801* 000000 000024
5830 000802* 001000 000376
5831 000803* 000000 000044
5832 000804* 001000 000310
5833 000805* 000000 000024
5834 000806* 001000 000376
5835 000807* 000000 000044
5836 000808* 001000 000310
5837 000809* 000000 000024
5838 000810* 001000 000376
5839 000811* 000000 000044
5840 000812* 001000 000310
5841 000813* 000000 000024
5842 000814* 001000 000376
5843 000815* 000000 000044
5844 000816* 001000 000310
5845 000817* 000000 000024
5846 000818* 001000 000376
5847 000819* 000000 000044
5848 000820* 001000 000310
5849 000821* 000000 000024
5850 000822* 001000 000376
5851 000823* 000000 000044
5852 000824* 001000 000310
5853 000825* 000000 000024
5854 000826* 001000 000376
5855 000827* 000000 000044
5856 000828* 001000 000310
5857 000829* 000000 000024
5858 000830* 001000 000376
5859 000831* 000000 000044
5860 000832* 001000 000310
5861 000833* 000000 000024
5862 000834* 001000 000376
5863 000835* 000000 000044
5864 000836* 001000 000310
5865 000837* 000000 000024
5866 000838* 001000 000376
5867 000839* 000000 000044
5868 000840* 001000 000310
5869 000841* 000000 000024
5870 000842* 001000 000376
5871 000843* 000000 000044
5872 000844* 001000 000310
5873 000845* 000000 000024
5874 000846* 001000 000376
5875 000847* 000000 000044
5876 000848* 001000 000310
5877 000849* 000000 000024
5878 000850* 001000 000376
5879 000851* 000000 000044
5880 000852* 001000 000310
5881 000853* 000000 000024
5882 000854* 001000 000376
5883 000855* 000000 000044
5884 000856* 001000 000310
5885 000857* 000000 000024
5886 000858* 001000 000376
5887 000859* 000000 000044
5888 000860* 001000 000310
5889 000861* 000000 000024
5890 000862* 001000 000376
5891 000863* 000000 000044
5892 000864* 001000 000310
5893 000865* 000000 000024
5894 000866* 001000 000376
5895 000867* 000000 000044
5896 000868* 001000 000310
5897 000869* 000000 000024
5898 000870* 001000 000376
5899 000871* 000000 000044
5900 000872* 001000 000310
5901 000873* 000000 000024
5902 000874* 001000 000376
5903 000875* 000000 000044
5904 000876* 001000 000310
5905 000877* 000000 000024
5906 000878* 001000 000376
5907 000879* 000000 000044
5908 000880* 001000 000310
5909 000881* 000000 000024
5910 000882* 001000 000376
5911 000883* 000000 000044
5912 000884* 001000 000310
5913 000885* 000000 000024
5914 000886* 001000 000376
5915 000887* 000000 000044
5916 000888* 001000 000310
5917 000889* 000000 000024
5918 000890* 001000 000376
5919 000891* 000000 000044
5920 000892* 001000 000310
5921 000893* 000000 000024
5922 000894* 001000 000376
5923 000895* 000000 000044
5924 000896* 001000 000310
5925 000897* 000000 000024
5926 000898* 001000 000376
5927 000899* 000000 000044
5928 000900* 001000 000310
5929 000901* 000000 000024
5930 000902* 001000 000376
5931 000903* 000000 000044
5932 000904* 001000 000310
5933 000905* 000000 000024
5934 000906* 001000 000376
5935 000907* 000000 000044
5936 000908* 001000 000310
5937 000909* 000000 000024
5938 000910* 001000 000376
5939 000911* 000000 000044
5940 000912* 001000 000310
5941 000913* 000000 000024
5942 000914* 001000 000376
5943 000915* 000000 000044
5944 000916* 001000 000310
5945 000917* 000000 000024
5946 00091
```

5602			15600	JNZ	NOTSTR	IF NOT VALTYP ALREADY=0
5603			15620	INR	A	IF(A)=1
5604			15640	STA	VALTYP	IFLAG THIS AS A STRING
5605			15660	RRC		IFAKE (A)=128
5606			15680	ADD	C	IFAKE (A)=SECOND CHARACTER
5607			15700	MOV	C,A	IFAKE INTO (C) WITH STRING BIT ON
5608			15720	CHRGST		IFGET CHARACTER AFTER "S"
5609			15740	NOTSTR:	>>	
5610			15760	IFN	LENGTH,<	
5611	006621	001000	15780	LDA	SUBFLG	IFGET FLAG WHETHER TO ALLOW ARRAYS
5612	006602	000000				
5613	006603	000000				
5614	006604	001000	15800	ADD	M	IFADD ONTO CHARACTER
5615			15820	CPI	"(	IFARRAY PERHAPS (IF SUBFLG SET NEVER WILL MATCH)
5616	006605	001000				
5617	006606	000000				
5618	006607	001000	15840	JZ	ISARY	IFIT IS!
5619	006610	000000				
5620	006611	000000				
5621			15860	IFN	LENGTH,<	
5622	006612	001000	15880	XRA	A	IFALLOW PARENS AGAIN
5623	006613	001000	15900	STA	SUBFLG	IFSAVE IN FLAG LOCATION
5624	006614	000000				
5625	006615	000000				
5626	006616	001000	15920	PUSH	H	IFSAVE THE TEXT POINTER
5627			15940	LENGTH=2,<		
5628	006617	001000	15960	MOV	A,D	IFVALUE TYPE INTO (A)
5629	006620	001000	15980	LHLD	VARTAB	IF(H,L)=PLACE TO START THE SEARCH
5630	006621	000000				
5631	006622	000000				
5632	006623	001000	16000	LOPFND:	PUSH	IFSAVE THE VALUE TYPE
5633	006624	001000	16020	XCHG	PSW	IF(D,E)=POINTER INTO SIMPLE VARIABLES
5634	006625	001000	16040	LHLD	ARYTAB	IF(H,L)=END OF SIMPLE VARIABLES
5635	006626	000000				
5636	006627	000000				
5637	006630	001000	16060	COMPAR		IFSEE IF THE END HAS BEEN REACHED
5638	006631	001000	16080	POP	H	IF(H)=VALTYP
5639	006632	001000	16100	JZ	NOTFNS	IFCOULDN'T FIND IT, SO MAKE A NEW ENTRY
5640	006633	000000				
5641	006634	000000				
5642	006635	001000	16120	LOAX	D	IFGET THE VALTYP OF THIS SIMPLE VARIABLE
5643	006636	001000	16140	MOV	L,A	IFSAVE SO WE KNOW HOW MUCH TO SKIP
5644	006637	001000	16160	CMP	H	IFCOMPARE WITH OUR VALTYP
5645	006640	001000	16180	INX	D	
5646	006641	001000	16200	JNZ	NOTIT1	IFNOT RIGHT KIND -- SKIP IT
5647	006642	000000				
5648	006643	000000				
5649	006644	001000	16220	LOAX	D	IF(A)=FIRST CHARACTER OF THIS VARIABLE
5650	006645	001000	16240	CMP	H	IFSEE IF OUR VARIABLE MATCHES
5651	006646	001000	16260	JNZ	NOTIT1	
5652	006647	000000				
5653	006650	000000				
5654	006651	001000	16280	INX	D	

5655	006652	001000	16300	LDAX	D	IFSEE IF SECOND CHARACTER MATCHES
5656	006653	001000	16320	CMP	B	
5657	006654	001000	16340	JZ	FINPTR	IFTHAT WAS IT, ALL DONE
5658	006655	000000				
5659	006656	000000				
5660	006657	001000	16360	XRD	"01000","076	IF"HV1 A," AROUND THIS INX SINCE THE POINTER
5661			16380			IFIS ALREADY INCREMENTED
5662	006660	001000	16400	NOTIT1:	INX	
5663	006661	001000	16420	INX	D	
5664	006662	001000	16440	MOV	A,H	
5665			16460			
5666			16480			
5667	006663	001000	16500	HVI	H,0	IFSKIP OVER THE
5668	006664	000000				IFCURRENT VARIABLE SINCE WE DIDN'T MATCH
5669	006665	001000	16520	DAD	D	IFSAVE THE VALTYP IN (A)
5670	006666	001000	16540	JMP	LOPFND	IF(H,L)=NUMBER OF BYTES TO SKIP
5671	006667	000000				
5672	006670	000000				
5673	006671	001000	16560	NOTFNS:	PUSH	IFSAVE THE LOOKS
5674	006672	001000	16580	MOV	C,H	IF(B,C)=LENGTH OF THIS VARIABLE
5675	006673	001000	16600	MOV	B,A	IF(B)=0
5676	006674	001000	16620	PUSH	B	IFSAVE THE VALTYP ON THE STACK
5677	006675	001000	16640	INX	B	IFMAKE THE LENGTH INCLUDE
5678			16660			IFTHE LOOKS TOO
5679	006676	001000	16680	INX	B	
5680	006677	001000	16700	INX	B	
5681			16720	IFN	LENGTH=2,<	
5682			16740	LHLD	ARYTAB	IFPLACE TO STOP SEARCHING
5683			16760	XCHG		
5684			16780	LHLD	VARTAB	IFGET THE PLACE TO START
5685			16800	LOPFND:	COMPAR	IFSEE IF WE ARE THERE
5686			16820	JZ	NOTFNS	IFCOULDN'T FIND THIS VARIABLE
5687			16840			IFSO MAKE ROOM FOR IT
5688			16860	MOV	A,C	
5689			16880	SUB	M	IFIS THIS VARIABLE THE ONE?
5690			16900	INX	H	
5691			16920	JNZ	NOTIT	IFNOPE
5692			16940	MOV	A,B	
5693			16960	SUB	M	IFTRY SECOND CHARACTER MATCHING
5694			16980	NOTIT:	INX	
5695			17000	JZ	FINPTR	IFTHAT WAS IT!
5696			17020	INX	H	IFSKIP OVER THAT ONE--NOT IT
5697			17040	INX	H	
5698			17060	INX	H	
5699			17080	INX	H	
5700			17100	JMP	LOPFND	IFTRY AGAIN
5701			17120	NOTFNS:	PUSH	IFREMEMBER WHAT THIS
5702			17140	LXI	B,6+SCODE	IFVARIABLE LOOKS LIKE
5703			17160			IFTHE AMOUNT TO SHOVE
5704			17180	LHLD	STREND	IFEVERYTHING UP BY
5705	006700	001000	17200			IFTHE CURRENT END OF STORAGE
5706	006701	000000				
5707	006702	000000				

Tommy's Simulator