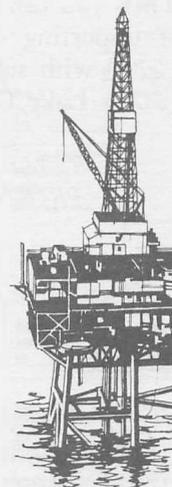


the I.P. Sharp newsletter

September/October 1979
Vol. 7 Number 5

IMPORTS

New Energy Data Base



Lib Gibson, Toronto

The IMPORTS data base contains the information published by the American Petroleum Institute in their monthly report entitled "Imported Crude Oil and Petroleum Products". The information originates with the Office of Oil Imports of the U.S. Department of Energy and is supplied to Sharp on computer tape by the API. The data is available on the SHARP APL time-sharing system approximately two months after the end of the reporting month and contains history back to January, 1977.

IMPORTS, one of the most detailed data bases available publicly in the petroleum field, has information on every shipment of crude oil or petroleum products into the U.S. You can find out, for instance, how much crude Company X imported in 1978 from Kuwait, through a particular port of entry, destined for Company T in some other city. Or you can crosstabulate, by country of origin, and by ranges of sulphur content, total shipments for the last year into each P.A.D. District (Petroleum Air Defence District). Or perhaps track over time Company A's imports of #2 fuel oil and forecast their future levels.

The facts stored about shipments are:

- date (month and year)
- commodity (crude, residual, unfinished or finished petroleum products)
- importing company
- port of entry
- country of origin
- recipient
- destination
- P.A.D. District - port of entry
- P.A.D. District - destination
- fee
- quantity in barrels
- sulphur content
- API gravity
- viscosity

IN THIS ISSUE

Imports data base	1	System Reliability	12
SNAP	3	Courses	13
Currency data base/Copenhagen	4		
EASy Actuarial system	5		
Chartered Banks update	5	Technical Supplement 22	
10th Anniversary of SHARP APL	6	Branching in SHARP APL	T1
SHARP APL at Sea	7	Solution to Summer Sum	T4
People	9	Contest 8 Results	T5
Applications Library update	11	Contest 9: Frequency Tabulation	T7

A special purpose language has been developed to facilitate report generation from IMPORTS. An example of its use is shown below. First you "load" the appropriate programs and variables:

```
)LOAD 121 IMPORTS
SAVED 13.43.40 8/15/79
```

Then you can enter a request for a report which essentially states "print a report showing full name of importing company (FIMPR), full name of country of origin (FORIG), and quantity imported (QTY), with subtotals (SUB) on quantity by company, for shipments of crude oil through port of entry (POE) Lake Charles, La. (code LAK) in January 1979".

```
TITLE 'CRUDE OIL IMPORTS THROUGH LAKE CHARLES, LA.'
PRINT(SUB FIMPR,FORIG,QTY) XIMC(POE EQL 'LAK'), (DATE EQL 1 79)
```

CRUDE OIL IMPORTS THROUGH LAKE CHARLES, LA.

<u>IMPORTER</u>	<u>COUNTRY OF ORIGIN</u>	<u>QTY/BBLs</u>
CITIES SERVICE OIL CO.	LIBYA	1,788,950
CITIES SERVICE OIL CO.	MEXICO	361,771
CITIES SERVICE OIL CO.	NIGERIA	19,834
CITIES SERVICE OIL CO.	SAUDI ARAB.	233,916
CITIES SERVICE OIL CO.	UN ARAB EM	263,151
CITIES SERVICE OIL CO.	UN KINGDOM	379,587
		<hr/>
CITIES SERVICE OIL CO.		3,047,209
		<hr/>
CONTINENTAL OIL COMPANY	ALGERIA	511,991
CONTINENTAL OIL COMPANY	LIBYA	385,674
		<hr/>
CONTINENTAL OIL COMPANY		897,665
		<hr/>
SHELL OIL COMPANY	ALGERIA	185,033
		<hr/>
TOTAL		4,129,907

Sharp's time series manipulation language, MAGIC, can also be used with the IMPORTS data base to provide crosstabulations, trend analyses, and forecasts over time. Since the access is compatible with access to private or other SHARP APL data bases, IMPORTS can be used in conjunction with other on-line data, in particular PETROSERIES. Anyone with access on the Sharp system who has a subscription to the API publication (\$200/year) can access IMPORTS.

This is an excerpt from an article which appears in the first issue of the **I.P. Sharp Energy Newsletter**, produced by the Sharp Energy and Natural Resources Support Group. Please contact your nearest SHARP APL representative if you want to subscribe, or if you want a copy of the manual "Imports Data Base" which will soon be available.

SNAP — SHARP NETWORK ANALYSIS PROGRAMS

Ken Chakawhata, London

You are currently involved in a project as project manager, coordinator, engineer, planner, foreman or committee member. You are familiar with and use Network methods as a project planning aid. Your project has been suitably broken down into appropriate activities whose completion is the realization of the project. The interrelationships between activities are portrayed in a network diagram which contains information about proposed activity execution sequences. Each activity in the network has had its duration estimated.

At the planning stage of the project, you will be faced with mainly exploratory questions:

- When should we start in order to meet our target completion date?
- When is the earliest we can finish given a specified start date?
- Given sufficient resources, when would each activity in the project be scheduled?
- Given limited resources, when can each activity be scheduled and how does this affect projected time limits?
- Is it worthwhile acquiring more resources?
- What is the effect of set time targets on resource requirements?
- What is the effect of resource limitations on time targets?

When the project actually starts executing you will want to know:

- Which activities have been completed?
- Which activities are currently in progress and who is responsible for them?
- How are we doing as compared to previous plan?
- Which activities are due in the near future?
- What is the likely impact of current status on future of the project?
- What modifications, if any, should we make to the current plan?

Network analysis (based on the critical path method) produces information which can go a long way towards giving an insight into the nature of the above questions. The basic computations involved in critical path analysis are desperately simple. However, for large projects with several hundred activities, it would be very difficult and time consuming, (hence expensive), to carry out the necessary network analyses without the aid of a computer. What you need then is a computerised system flexible enough to cope with the dynamic nature of project planning and control.

SNAP IS SUCH A SYSTEM!

There are many such project network analysis programs on the market today. So, what does *SNAP* have to offer?

SNAP is written in a very powerful language, SHARP APL. This fact has allowed us to develop an equally powerful project planning and control language that enables you to:

- specify your own data input and output format
- select output data for reports
- define networks suitably broken down into subnetworks to reflect work breakdown structure or company organization.

Buried in the above general facilities is quite a wide range of network data processing functions. However, *SNAP*'s modular design means that you need only concern yourself with functions that are relevant to your particular needs. In fact you can get by without even knowing of the existence of some *SNAP* functions.

A system that boasts every possible facility is likely to be too complex and cumbersome to be usable. *SNAP* does not and need not. The system design is open ended enough to allow additional customized features to be interfaced easily.

We have tried to make *SNAP* similar to SHARP APL while at the same time allowing you to use *SNAP* without any prior knowledge of SHARP APL. In fact, *SNAP* and APL coexist quite happily so that, although you may not be aware of it, you will have the full power of APL available should you require it.

You drive *SNAP*, not the other way round, but *SNAP* allows you to program it to drive you if you want it to. By using *SNAP* building blocks, you can design your own macro *SNAP* to suit your operational environment.

The most important asset of the *SNAP*-APL environment is that *SNAP* can communicate very easily with other SHARP APL based systems, e.g manpower planning, cost control, financial analysis or general data base systems. This means that *SNAP* can easily supply data to or receive data from other SHARP APL systems. It is therefore possible to design an integrated but loosely coupled project management information system whose components can be interfaced at will.

SNAP can be accessed from the I.P. Sharp public library workspace 48 *SNAP*. Contact your local Sharp office for a "snappy" demonstration.

COPENHAGEN MARKET ADDED to the CURRENCY EXCHANGE RATES DATA BASE

Johnny Larsen, Copenhagen

The currency data base has been extended to include data from Copenhagen. The new market consist of 122 time series, updated on a daily basis by the Copenhagen staff.

The data may be categorized as follows:

- 1 noon spot rates
- 2 opening rates, buy and sell
- 3 forward rates, buy and sell for 1, 3, 6 and 12 months.

All the series are not included in all three groups:

- 1 2 3 contains data for USA, U.K., GER, SWE, NOR, NTH, SWI, JAP, BEL, FRA.
- 1 2 contains data for SPA.
- 1 contains data for AUS, FIN, CAN, ICE, IRE, POR, ITA.

The buy and sell quotations are from the banks' point of view. The prices are expressed in Danish ore. A U.S. spot rate at 525 shows that it costs 525 ore for U.S. \$1. The forward rates are expressed as a premium to either add to or subtract from the opening rates.

Access to the Copenhagen currency exchange rates may be done either via 39 *MAGIC*, or 81 *CURRENCY*. For further information please contact me directly (in the Copenhagen office), or your local SHARP APL representative.

EBSCO ACTUARIAL SYSTEM = EASy

David Crossley, EBSCO, Toronto

EASy is an actuarial system developed in SHARP APL by Eckler, Brown, Segal & Company, Consulting Actuaries. EASy makes the following calculations in accordance with the 1978 Canadian Life Insurance regulations:

1. **Statement Reserve Factors**
2. **Maximum Tax Reserve Factors**
3. **Calendar Year Projections of Profits for a unit of insurance.**

EASy is totally user-oriented, and does indeed live up to its name. It is particularly useful for testing the effect of different assumptions on reserves and profits, but can also calculate reserve factors for all issue ages in the rate book.

For more information, and charges, contact D.W.F. Crossley at:
Eckler, Brown, Segal & Company Ltd.,
789 Don Mills Road,
Don Mills, Ontario, M3C 1T8
(416) 429-3330

CHARTERED BANKS DATA BASE UPDATE

Cathy Calvert, Toronto

The Continental Bank of Canada has been added to the Monthly, Quarterly and Annual Canadian Chartered Banks data bases. Monthly data is available from June 1979. Quarterly and Annual data will be available from the period ending October 31, 1979.

To access data for this new bank (13) and the two pseudo banks (101 and 102: see the May/June '79 newsletter 7.3), please use the **updated** access functions in library 54. Users who have previously saved copies of the Chartered Banks access functions are advised to copy in the new versions available in library 54 to access the new data.

A DECADE OF APL!

On August 1st, 1969, just over 10 years ago, APL went live from Toronto, connecting a small group of enthusiasts in two locations in the city with friends in Ottawa, Waterloo, Windsor, Hamilton and Bethesda. In July of this year, SHARP APL became available on board a luxury ocean liner via satellite (see page 7):

1969

From the operator log during those first weeks:

A P L \ 3 6 0 NOTES 07/20/69

07/20/69 THE PROBLEM WITH SQUARE ROOT HAS BEEN ELIMINATED.
THIS TWO-WEEK PERIOD IS FOR IMPROVEMENTS TO THE SYSTEM AS
WELL AS FREE CUSTOMER EDUCATION. ANY PROBLEMS SHOULD BE REPORTED
TO THE OPERATOR WHEN DISCOVERED.

10.10.53 07/24/69 WAPSIT

032:R VICTORIA OFFICE HERE VIA HAMILTON TIELINE VIA WATERLOO TIELINE
)MSGN 32 WHAT A WAY TO GO!

OPR) 9.17.51 07/31/69 OPERATOR
)HI GOOD MORNING. APL WILL RUN TIL 9:00 P.M. TONIGHT.
MAKE THE BEST OF THE LAST FREE DAY!

036:RHOW MANY BYTES TO A WORKSPACE?
)MSG 36 32E3

OPR) 8.27.58 08/01/69 OPERATOR

)HI GREETINGS AND SALUTATIONS ON THIS THE FIRST
OFFICIAL DAY OF APL\360 SERVICE

044) 16.25.01 08/08/69 ...
)MSG 44 HURRAY YOU'RE ON

044:RTHIS IS WASHINGTON BASE

003 17.07.06 11/12/69
003:RWHAT WAS THE PROBLEM THAT CAUSED THE SYSTEM TO GO CRASH
THIS AFTERNOON?

)MSG 3 RHL+LMB+2821

1979

Telex printout on the QE2, June '79:

FFL DE GBTT BONJOUR ERE CALL TO HOLLAND

NUMBER IS 18859 AB IS ITS NL +?

TXMAR A STLYS F
+?.
MOM
OK GA +?

18859 ITS NL+?(
CUNARD-QE2.

0)
)#####
3744) 12.01.21 06/17/79 CHIVERS

SHARP APL SERVICE

)LOAD 666 BOX
SAVED 19.46.12 06/02/79
SEND

TO FJP
TEXT:
SENT FROM THE QE2 IN SOUTHAMPTON VIA STLYS
RADIO, FRANCE BY SITOR.

ACTION: CC JTAY IPS
ACTION: SEND
NO. 771591 FILED 12.04.01 SUN 17 JUN 1979
TO

SHARP APL AT SEA

David Chivers, London

So you thought you could get away from it all by spending a leisurely few months on a QE2 World Cruise? Sadly, that's no longer true, at least if you are a SHARP APL user. Interactive access to SHARP APL is available from the QE2, or indeed from most large cargo or passenger ships, or from oil rigs, allowing you to access your company's data or the public data bases, to use mailbox, or even to add 2+2, wherever the ship may be.



By courtesy of Cunard line, James Taylor (the freelance consultant behind the idea) and I boarded the QE2 in Southampton, where we were instructed by the Chief Radio Officer, Mr. R. Butterworth, in the mysteries of ship-to-shore radio, and in operating your own satellite earth station.

We signed on to SHARP APL by two different routes, both connecting to our network via Telex access points.

Conventional ship to shore radio telex uses a half-duplex error-corrected path between ship and shore. Placing a call involves setting the appropriate radio frequencies to call up a coastal radio station, then requesting the coastal operator to place a Telex call to the Sharp access number. He responds with a 'Go ahead' when the call is connected; after that, you sign on almost as normal — the difference is that every time you finish typing, and expect APL to reply, you have to press an 'over' button to reverse the direction of transmission, or else you receive nothing. Similarly, when APL has finished printing, 'over' is again required to allow you to type. Fortunately there's a 500-character buffer at each end, so if you are a bit slow in pushing the 'over' button, you won't lose any characters.



Ship-to-Shore Radio Telex

If the radio path is noisy, there will be significant pauses in printing while correction of the erroneous characters takes place. The cost of a ship-to-shore call is about \$1/minute, and the additional cost of a long distance call can usually be avoided by calling a radio station close to one of our Telex access points.



← The Bridge

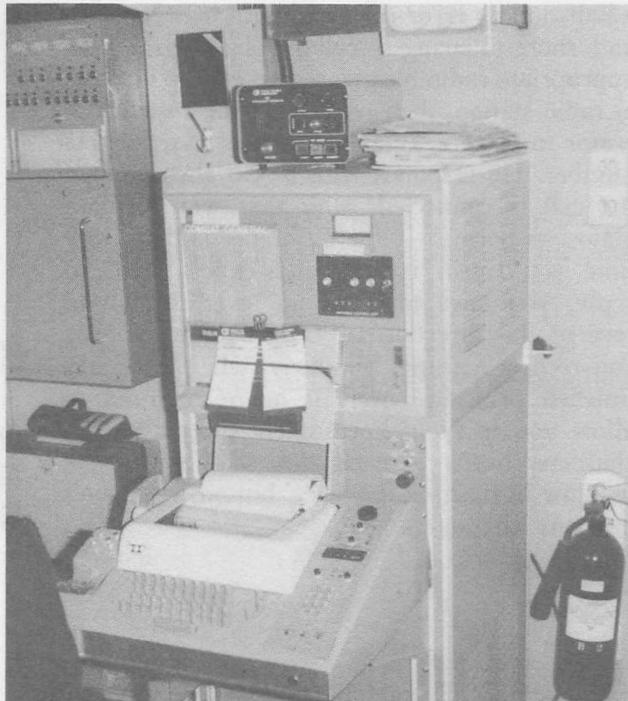
Alternatively, larger ships are equipped with a directional aerial to pick up one of three communications satellites serving the major ocean areas. The shipboard equipment is capable of carrying voice telephone traffic at \$13/minute, or Telex traffic at \$4/minute.

The system is automatic, just press the 'call' button and type the SHARP APL Telex access number, and within seconds the call is connected, with a good quality line free of character errors.

Is Sunday, 17th June 1979 a date for the history books? It may well be, for we can find no evidence of any previous attempt to access a commercial time-sharing computer system from a ship.

My only regret is that the experiment was successfully concluded so quickly that we had no excuse to remain on board for the next voyage to New York!

MARISAT →



Technical Supplement-22

BRANCHING IN SHARP APL

Robert Metzger, IPSA Rochester

APL is unique among high-level programming languages in a number of respects. One of the most significant differences is that APL provides a single function, the branch arrow, to change the order in which statements within a function are executed. The branch arrow is a potential source of trouble to all APL users if it is misused, but it tends to be a particular problem to people who are used to other programming languages, or who are not professional programmers. This article begins a series on how to avoid unnecessary use of the branch arrow, and on how to use it properly when the situation warrants it.

What is the branch arrow used for? Since the statements in an APL function are executed one after another from the beginning to the end, the branch arrow makes it possible to execute lines in some other order. Fundamentally, there are only two ways to change the order of execution:

1. To choose one set of statements instead of another, based upon some condition, or
2. To do a set of statements repeatedly while some condition is true.

The default order of execution, which APL provides, is SEQUENTIAL, while the other two types are called ALTERNATIVE and ITERATIVE. All patterns of branching fall into the ALTERNATIVE or ITERATIVE categories.

Why would we want to avoid branching? Branching complicates the simple sequential order of execution of an APL function, and increases the number of statements required. Anything which makes a function longer or more complicated makes it more susceptible to programmer errors. In addition, in APL, solutions to problems which do not require explicit looping (ITERATIVE structures built with branch statements) often use considerably less CPU time to execute.

How can we avoid branching? There are at least 8 different methods available. The first method is:

USE BOOLEAN CONDITIONAL STATEMENTS

A Boolean Conditional statement is one which takes advantage of the fact that the Relational ($< \leq = > \geq \neq$) and Logical ($\wedge \vee \neg \forall$) primitives in APL are functions, rather than assertions, which return boolean results (0's and 1's) which can be used by the other APL functions.

Boolean conditional statements use the special meanings that 0 and 1 have for certain primitive functions to avoid branch statements. We can take advantage of the fact that multiplication and division have the identity element 1 to compute statements of the form

If test, Then $X \leftarrow NUMBER$, Else $X \leftarrow 0$.

This might be written with branches as the following:

```
IF:→(∼TEST)/ELSE
THEN: X←NUMBER
→IFEND
ELSE: X←0
IFEND:
```

With a boolean conditional, it is a single statement:

$$X \leftarrow \text{NUMBER} \times \text{TEST}$$

An example of the use of such a statement would be to search for a character in a vector, but to return a 0 (instead of 1+ρ VECTOR) if it was not found.

$$\text{POSN} \leftarrow (\text{VECTOR} \setminus \text{CHAR}) \times \text{CHAR} \in \text{VECTOR}$$

Another way to use boolean conditionals is to write statements of the form:

If test, Then $X \leftarrow \text{NUMBER}$, Else $X \leftarrow -\text{NUMBER}$.

This might be written with branches as:

```
IF: →(∼TEST)/ELSE
THEN: X←NUMBER
→IFEND
ELSE: X←-NUMBER
IFEND:
```

but with a boolean conditional it is simply:

$$X \leftarrow \text{NUMBER} \times \bar{1} * \text{TEST}$$

Here we take advantage of the fact that the power function returns a 1 when exponentiating to the 0th power, and of the identity element of multiplication.

An example of the use of such a statement would be:

$$\text{PART} \leftarrow (5 \times \bar{1} * \text{TEST}) \uparrow \text{VECTOR}$$

which takes the first 5 or last 5 elements from the vector, depending on the test.

A third way to use boolean conditional statements is to write statements of the form

If test, Then $X \leftarrow \text{NUMBER}_1$, Else $X \leftarrow \text{NUMBER}_1 + \text{NUMBER}_2$.

This might be written with branches as:

```
IF:→(∼TEST)/ELSE
THEN: X←NUMBER1
→IFEND
ELSE: X←NUMBER+NUMBER2
IFEND:
```

but with a boolean conditional it is simply:

$$X \leftarrow \text{NUMBER1} + \text{NUMBER2} \times \text{TEST}$$

In this case, we use the identity element of addition and subtraction (0) and the identity element of multiplication as well.

An example of the use of such a statement would be the following function, which catenates or laminates vectors and matrices along the first dimension. The computation of the axis is done with a boolean conditional.

$$\begin{aligned} & \forall \text{MATRIX} \leftarrow \text{TOP JOIN BOTTOM}; \text{WIDTH} \\ [1] & \text{WIDTH} \leftarrow (\bar{1} \uparrow \rho \text{TOP}) \uparrow (\bar{1} \uparrow \rho \text{BOTTOM}) \\ [2] & \text{MATRIX} \leftarrow ((\bar{1} \uparrow \rho \text{TOP}), \text{WIDTH}) \uparrow \text{TOP}, \\ & \quad [\square \text{IO} - .5 \times (1 = \rho \rho \text{TOP}) \wedge (1 = \rho \rho \text{BOTTOM})] \\ & \quad (\bar{1} \uparrow \rho \text{BOTTOM}), \text{WIDTH}) \uparrow \text{BOTTOM} \end{aligned}$$

A final type of boolean conditional implements statements of the form:

$$\text{If test, Then } X \leftarrow \text{ARRAY}[\dots; 1; \dots] \text{ Else } X \leftarrow \text{ARRAY}[\dots; 2; \dots]$$

This might be written with branches as the following:

```
IF: → (~TEST) / ELSE
THEN: X ← ARRAY [ ...; 1; ... ]
→ IFEND
ELSE: X ← ARRAY [ ...; 2; ... ]
IFEND:
```

Using a boolean conditional it is simply:

$$X \leftarrow \text{ARRAY}[\dots; \square \text{IO} + \text{TEST}; \dots]$$

This expression makes use of the fact that adding the index origin to 0 or 1 always computes a valid origin independent index.

An example of the use of such a statement would be:

$$\text{MATRIX} \leftarrow \text{MATRIX}[\Delta \text{MATRIX}[\dots; 2 \ 4[\square \text{IO} + 1 = \text{FLAG}]]];]$$

which sorts the rows of a matrix according to the values in either columns 2 or 4, depending on whether *FLAG* is 0 or 1.

This section has shown a number of ways in which boolean conditionals can be used to replace ALTERNATIVE branching structures. The benefits realized from using such statements are smaller, simpler programs and consequently less time required to write and debug programs. In subsequent articles, other methods of avoiding branching, such as subroutines, recursion, and the execute function, will be explained.

Solution to Summer Puzzle

The clue to the solution is that the total flight of the plane going to Lisbon and then to Madrid is over 20.5 hours, and at a rate of 1000 km/hr this makes a distance of over 20500 km. Since the earth's semicircumference is only 20000 km, it should have been clear that the first plane flew more than halfway around the globe.

The great circle (shortest) routes from Wellington to Lisbon and from Wellington to Madrid are in opposite directions. Flying to Lisbon a plane would follow a generally northeast route; to Madrid it would fly northwest. The distances are:

	LISBON	MADRID
WELLINGTON	19583	19858
LISBON		501

So the total distance Wellington-Lisbon-Madrid is 20084 km, or more than half-way around the globe, as against a distance of only 19858 km flying direct to Madrid, or 226 km shorter. 226 km can be flown in 13.5 minutes, at 1000 km/hr, thus the second plane, even though taking off ten minutes later, will fly 13.5 minutes less, and be on the ground more than three minutes before the first plane.



The following map shows the earth in a Mercator projection, locating just Wellington, Lisbon, and Madrid:



*** Contest 8 Results ***

Congratulations to all who took part in our eighth contest. The entrants and their solutions were:

Roger Hui, Edmonton	<i>EDITΔHUI1/2/3/4/5/6</i>
Kevin Harrell, Newport Beach	<i>EDITΔKEV1/2/3/4</i>
Stuart Baker, Gloucester	<i>EDITΔSTUB</i>
Josh Levine, Chicago	<i>EDITΔJSL</i>
Dr. Sashi Date, Trumbull, Ct.	<i>EDITΔSHA1/2</i>
Ernie Bollo, CPAir, Vancouver	<i>EDITΔBOL</i>
Rohan Jayasekera, Montreal	<i>EDITΔROHAN</i>
David Fiori Jr., Yardley, Pa	<i>EDITΔFIORI</i>
Joe Trivers, Toronto	<i>EDITΔJMT</i>
M.E. Martin, Wembley	<i>EDITΔMEM1/2/3</i>
David Church, Garland, Tx	<i>EDITΔDAV</i>
Steve Cooper, Toronto	<i>EDITΔSCOOP</i>
Henri Brudzewsky, Copenhagen	<i>EDITΔHENRI1/2/3/4</i>
Wolfgang E. Fendt, Heidelberg	<i>EDITΔWEF1/2</i>

When tested for completeness, all but five solutions satisfied the minimum requirements of the contest. Most of those who submitted extension entries did so by submitting two or more entries. These extra solutions were not included in the timing tests, since they were assumed to be slower than the stripped-down versions. The remaining entries were then submitted to timing tests. The execution times, expressed as multiples of the fastest entry, are tabulated as follows:

Entry	Ranked Time
<i>EDITΔHUI6</i>	1.000
<i>EDITΔHUI5</i>	1.122
<i>EDITΔHUI4</i>	1.123
<i>EDITΔHUI2</i>	1.172
<i>EDITΔHUI3</i>	1.174
<i>EDITΔHUI1</i>	1.239
<i>EDITΔROHAN</i>	1.345
<i>EDITΔJMT</i>	1.451
<i>EDITΔSTUB</i>	1.512
<i>EDITΔWEF1</i>	1.537
<i>EDITΔDAV</i>	1.535
<i>EDITΔMEM1</i>	1.676
<i>EDITΔBOL</i>	1.842
<i>EDITΔFIORI</i>	1.850
<i>EDITΔHENRI3</i>	3.829
<i>EDITΔHENRI1</i>	3.842
<i>EDITΔJSL</i>	4.549 *
<i>EDITΔHENRI2</i>	6.353
<i>EDITΔHENRI4</i>	6.367
<i>EDITΔSHA1</i>	6.882
<i>EDITΔSHA2</i>	6.987

*meaningful timings not possible since this solution (the only one submitted) contained extensions.

Notes

1. This testing was done on typical lines of code of length 120.
2. *EDITΔWEF1* performed most efficiently on shorter strings (e.g. length 69).
3. *EDITΔHUI6* performed most efficiently on much larger strings (e.g. length 500)

The overall winner of Contest 8 is *** **Roger Hui** ***, of our Edmonton office. His final entry, *EDITΔHUI6* was, to say the least, compact — one line! Our non-IPSA winner is the overall winner of our last contest, *** **Wolfgang E. Fendt** ***. As usual, his solution was concise and a pleasure to read. The winners will be contacted shortly regarding their prizes.

Since many of the entries were fast, a large number of them will be placed into workspace 999 *CONTEST*. Many thanks to all who participated.

The following entries proposed interesting extensions to the line editor (and may also be found in 999 *CONTEST*): *EDITΔWEF2*, *EDITΔMEM2*, and *EDITΔJSL*. Brief descriptions of these extensions follow:

1. *EDITΔWEF2* — Mr. Fendt's 'cross-mode' extension.

If the character \times appears as the first element of the left argument *CONTROL*, his function will add it optically to the right argument *TEXT*. For example, if

TEXT ← ' _O ABC'

CONTROL ← 'xA*-_XC'

the result would be ↔ 'A⊕-AXC'

Mr. Fendt also notes that 1) characters in *TEXT* which are not compatible with those in *CONTROL* at the same position would be replaced by an overlay character of *CONTROL*; and 2) unfortunately, the *CONTROL* string is shifted one position to the right because a control character is needed.

2. *EDITΔJSL* — Mr. Levine calls his extensions ω and α . These go into the *CONTROL* argument. They are best described as follows:

Command	Syntax	Description
ω	$\omega\{\text{string1}\}\{\text{string2}\}$	replaces string1 with string2 . Acceptable delimiters are <> ⇄ { } c⇒ ∩ ∪ ∇ Δ () [] .
α	αstring	appends string to end of <i>TEXT</i> .

3. *EDITΔMEM2* — Mr. Martin has made an extension to allow multiple insertions on one line. Thus, with a little forethought, editing may be done in one pass every time (almost). This necessitates the use of a delimiter. He has chosen \vdash on the grounds that it is 'lower case, meaningless, and graphically suitable'. example, if

TEXT ← 'THE BROWN DOG ATE THE GREEN APPLE AND DIED'

CONTROL ← ' // // // ,STUPID- // .ON- // / .AR'

the result would be

'THE STUPID DOG ATE ONE GREEN PEAR AND DIED'

He notes that 1) the delimiter may be elided in the last insertion; and 2) the only drawback is that the delimiter itself can not be inserted directly, but can, of course, be done in the old-fashioned way.

*** CONTEST 9 *** FREQUENCY TABULATION ***

Mike Holloway, Toronto

The following problem lends itself nicely to applications in the actuarial, financial and manufacturing fields.

In order to direct their advertising campaign effectively, a brewery needs information on the habits of the average beer drinker. To tabulate the number of bottles consumed per week, broken down by age of the drinker and the type of beer, their consultants (a marketing service company) place questionnaires in every outlet. The questionnaires have only the following three questions:

1. Do you drink ale, lager, or stout most often? (1/2/3)
2. How old are you?
3. How many bottles do you drink per week?

Note that they did not favour a multiple-choice questionnaire, but one that could result in large numbers of different answers. Thus they will have to make groupings on their report for the brewery.

Suppose that they decide on the following format:

ALE

	Bottles per Week				
Age	0-3	4-12	13-25	25-36	more than 36
18-22					
23-27					
28-35					
36-45					
45-65					
over 65					

with identical planes for LAGER and STOUT.

Required: A $\square IO$ -localized function with syntax:

$R \leftarrow CONTROL FTAB DATA$

For example, given the above, the result of the function would be a rank-3 array of shape $\leftrightarrow 3 \ 6 \ 5$. The arguments can be described as follows:

DATA - will be a numeric, rank-2 array with shape $\leftrightarrow M \ N$ where M is the number of questions on the questionnaire and N is the number of questionnaires (i.e. the sample size). Thus, $DATA[I;J]$ will be the answer to question I on questionnaire J . In the above example, M would be identical to 3 and N could be arbitrary.

CONTROL - will be a numeric, rank-2 array with shape $\leftrightarrow 2 P$, where P is relatively arbitrary. The first row of *CONTROL* will be a Boolean partition vector, with ones designating the start of the partitions and zeroes elsewhere. The simple identity holds that:

$$M \leftrightarrow +/CONTROL[\square IO;]$$

or in other words, there will be M partitions represented within *CONTROL*. Thus, in the example given there will be three ones. The second row of *CONTROL* contains the vectors of lower bounds for the groupings desired. For the example given, *CONTROL* would be:

```
1 0 0 1 0 0 0 0 0 1 0 0 0 0
1 2 3 18 23 28 36 46 66 0 4 13 25 37
```

Thus, the first partition holds the lower bounds for ranges derived for the first question, the second for the second question, etcetera.

Note that the partition vector is only there to provide the function with information it needs to split up the range lower bounds in the second row. Note also that the following condition must always hold:

$$\rho R \leftrightarrow (1+T)^{-1} + T \left(CONTROL[\square IO;], 1 \right) / \iota P + 1$$

where P is as above and T is a temporary.

Example 2: Let

```
CONTROL ↔ 1 0 1 0 0
           -1 2 0 2 4

DATA ↔ -1 -1 1 -1 -1 2 15 2
        1 2 3 7 2 17 64 1
```

then the result of *CONTROL FTAB DATA* would be a rank-2 numeric array of shape 2 3 as follows:

```
1 3 1
1 0 2
```

Criteria for Evaluation: As usual both processor and workspace efficiency will be measured, and your entries rated accordingly.

Submissions: \square PACK your function, with a variable named *AUTHOR* containing your name and address. \square APPEND this package to file 999 *CONTEST* no later than December 31, 1979. Good luck!

Prizes: A prize of \$50 cash (Canadian) will be awarded to the best non-IPSA entry, while the best entry from I.P. Sharp Associates will receive a book prize of the author's choice.

SHARP PEOPLE

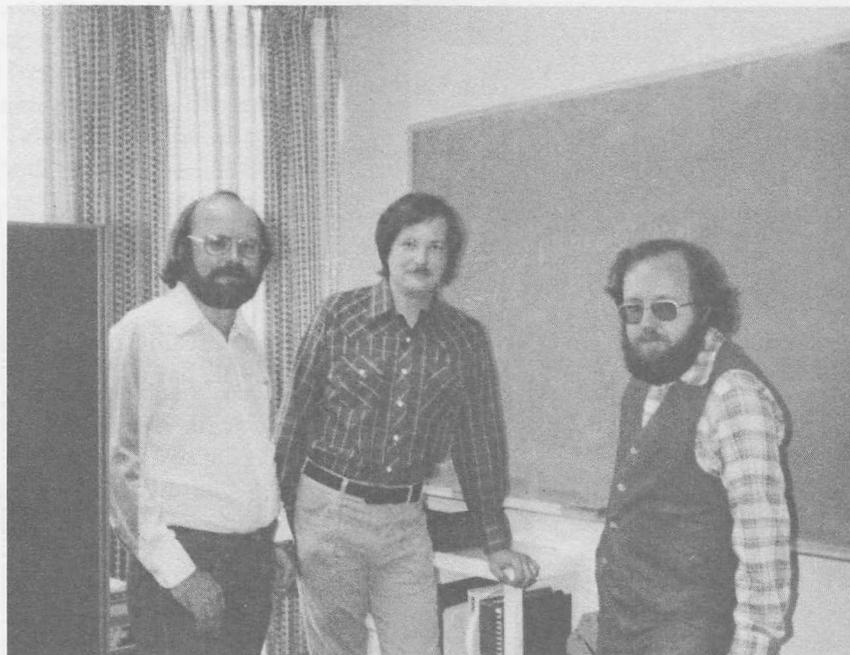
ROCHESTER

The Rochester branch of the I.P. Sharp Software Development Group, more familiarly known to Sharp insiders as Zoo, grew from one member to three as of July 1. Headed by Eric Iverson in Toronto, Zoo is now in three offices: Toronto, Palo Alto, and Rochester. The members of Zoo seek to enhance and maintain APL within the Sharp system and to provide maintenance and support.

Together **Gary Dennis**, **Mark Dempsey**, and **Charles Chandler** (Chan) of the Rochester Zoo have nearly twelve years of experience within IPSA plus many more years with various other APL users. Gary joined Zoo in the fall of 1974 after a year and a half as an APL consultant/operations programmer. Presently, he is readying a new in-house SHARP APL release for a customer. Although Mark is new to Zoo, he is no stranger to Rochester IPSA. He has worked both in Special Systems and as an APL systems consultant. Mark is very enthusiastic about his move to Zoo and is already involved in an in-house APL installation at a nearby university. The third member, Chan, joins Zoo after several years as a system programmer at Xerox. Consequently, he knows first hand what kind of support distributed SHARP APL customers need and will work with Gary and Mark to provide this service. He is currently involved in developing performance measurement tools. This combination of experience gives Rochester a valuable and skilled team of experts to initiate the Zoo branch.

Although the Software Development Group is officially headquartered in Toronto, there is a special cohesiveness among all its member whether they are in Rochester, Palo Alto, or Toronto. They are in constant communication with each other through the on-line mailbox system and by phone. In addition the Rochester members often attend the weekly Toronto Zoo meetings thereby continuing the on-going exchange of ideas, information, and coordination of projects. Zoo members are encouraged to come to Toronto to use the facilities and equipment there for testing and development. However, a growing collection of remote development and testing tools makes it possible to perform most Zoo functions without being physically present in Toronto.

Gary Dennis, Mark Dempsey and Charles Chandler



TORONTO UPDATE

Ted Talbot

A number of new people have joined the Toronto Branch since its move to larger premises in December, 1978.

For the past two years **Brian Olson** was the merchandising Buyer for the Canadian distributor of an imported automobile line. He was responsible for researching and developing marketing programs. Now that Brian has joined Sharp, he will be devoting his skills to financial applications, including the new leasing package. Brian has a Bachelor of Commerce degree from the University of Toronto.

Jim Spurgeon joined Sharp with three years experience in the insurance and re-insurance business. He was also a Mathematics lecturer at the University of Western Ontario where he obtained an M.A. in Actuarial Science. Jim is a member of the American Statistical Society and has completed the first three parts of the Society of Actuaries examinations. With his two years of APL experience, he is already quite actively involved in augmenting our services to the insurance industry.

After spending two work terms with us as a Co-op student, **Phyllis Nishikawa** graduated (B.Math, University of Waterloo) and joined the company on a full-time basis. She is currently providing support to our Banking clients, and is responsible for maintaining several related planning systems.

We are also pleased to announce the addition of the following recent graduates who will be involved in customer support: **Chrysanne di Marco, Kathryn Gooding, Lucy Gorny, Timothy Humble, Karen Makarchuk, Joan Templeton, and Brian Traquair.**

"Who's Who" en France

Tim Cureton

La filiale parisienne d'I.P. Sharp accueille deux nouvelles recrues.

Uriel Wittenberg vient du bureau de Toronto. Uriel, de nationalité française, a passé un certain temps à Montréal, New York et Toronto où il s'occupait de l'enseignement de l'APL et des applications dans le domaine de rapports statistiques.

Malcolm Hyde a décidé d'entrer dans la compagnie, ayant déjà travaillé quelques mois à Paris. Il a travaillé à Bruxelles et à Paris sur les applications de l'aviation. Il est, par ailleurs, en train de terminer sa licence d'informatique.

APPLICATIONS LIBRARY UPDATE

Ed Stubbs, Toronto

The significant changes in the applications library workspaces since March are:

{(n) - new, (c) - changed, (m) - moved, (r) - removed}

- 2 *DIRECTDEF* (n) - allows definition of functions using the direct method of Iverson and Orth
- 9 *SCANDOC* (m) → 2 *SCAN* (n)
- 13 *NATIONS* (c) - addition of function to calculate distances using oblate spheroid model of the earth
- 21 *GENERAL* (c) - assorted mathematical functions, new improved version
- 26 *POLYNOMIALS* (n) - symbolic manipulation of polynomials, which may be multi-variate
- 32 *NPE* (n) - nonlinear parameter estimation
- 51 *FPREPORTS* (n) - produces historical financial reports for companies in the Financial Post fundamental data base
- 51 *NEWTSEINDEX* (r) - data now available from 55 *RETRIEVE*
- 51 *STOCKOPTION* (r) - data now available from 55 *RETRIEVE*
- 55 *RETRIEVE* (n) - for retrieval of options, commodities and Toronto Stock Exchange Index data
- 81 *CSUSAGE* (c) - new function *INQUIRE* to inquire on available series
- 95 *RECIPES* (n) - accesses a data base of recipes
- 85 *MABRAUTIL/86 MABRA* (c) - new version, see descriptions of enhancements in the workspaces
- 121 *IMPORTS* (n) - oil imports data base access workspace (see page 1)

SYSTEM RELIABILITY

Bill Apsit

On-line information that can help you to monitor your usage of SHARP APL, and to maintain the security and integrity of your systems and data, is available. The workspace 999 *RELIABILITY* contains monthly summaries of the SHARP APL system uptime for the central computer site. We update it from time to time, and it has summaries for past years.

The uptime an individual user experiences is somewhat less than that of the central computer site. Additional outage time results from failures in the communications network connecting a user to SHARP APL. Though the IPSA network is highly reliable, your local connection to the IPSA network may not be. This additional outage time varies greatly from place to place. Partial outages at the central computer site also affect user uptime. An example of this is a failure of the 'slave' CPU which 'crashes' two users while leaving the other 200 undisturbed.

Though outages are often a nuisance, we should realise that they do occur and consider them in system planning. It is important that a system can restart or rerun cleanly after an outage. A backup method of connecting to the IPSA network is valuable when the ordinary method fails. It is important that you evaluate your requirements, and ensure that a lengthy outage would not be utterly catastrophic. Your local SHARP APL representative can explain the standard system backup and offsite storage facilities. If you wish, you can arrange for additional backup of your critical systems, and store them offsite independently.

In previous years SHARP APL shut down daily, most recently for two hours per day during our weekday schedule.

This year our weekday schedule is for uninterrupted operation from 0400 UTC each Monday until shutdown 0400 UTC Saturday for our weekly fulldump. We have also scheduled significantly more hours of service on weekends, but the weekend schedule is subject to change at short notice. The following summary of availability includes only weekdays, with the exception of New Year's Day. We have on a number of occasions scheduled weekday shutdowns for urgent repair or testing. Each of these is considered an outage in this summary. The only exceptions to this are New Year's Day and Christmas Day.

Month	Outages	Mins. out	% Up-time	Weekdays
January	8	71	99.8	22
February	5	97	99.7	20
March	3	98	99.7	22
April	8	145	99.5	21
May	20	330	99.0	23
June	10	109	99.6	21
July	12	156	99.5	22
August	9	106	99.7	23
Year-to-date	75	1112	99.6	175
Mean time between outages:		56 hours		
Mean time per outage:		15 minutes.		

COURSE SCHEDULE

SEMINARS and SPECIAL COURSES

Birmingham

Formatting with SHARP APL, November 30
 Plotting Facilities Available on SHARP APL, October 31
 Shared Variables - A New System Feature, September 28
 Using SHARP APL Files, December 17

Gloucester

Appreciation of APL, November 5
 Security, September 24

London

Advanced APL, October 15, November 29
 APL Review, October 22, November 21, December 12
 Appreciation of APL, October 19, November 9, December 3
 CONSPACK - A Consolidation Package on SHARP APL, November 26
 Data Bases Available thru SHARP APL, October 26
 Linear Programming with SHARP APL, November 2
 MAGIC for Time Series Analysis, October 4
 SNAP - An Intr. to a Network Analysis Pkg., November 28
 The Commodities Data Base on SHARP APL, October 26
 Use a Terminal, October 2

Montreal

AIDS, October 16-18 (English or French)
 Applications of Project Planning and Control Techniques, October 11 (English)
 Error Trapping, October 23 (English), November 20 (French)
 Planning Systems Design, November 8 (English)
 Plotting, October 30 (English)
 Shared Variables, November 22 (English), December 4 (French)
 Saving Money with N&B-tasks, October 10 (English), November 6 (French)

Ottawa

AIDS, February 19
 Batch (Non-Terminal) Tasks, December 18
 Box-Jenkins, October 23
 Highspeed Printing, November 22
 Input Validation, October 25
 MAGIC, December 20
 Plotting, February 21
 Regression Analysis, November 20

San Francisco

Graphics, November 6

Toronto

Actuarial APL Techniques, Oct. 9, Dec. 10
 AIDS - Introduction, December 4-5
 APL for Managers, November 23
 Box-Jenkins, November 8
 Data Base Design, October 19, December 20
 Forecasting Methods, November 15
 Plotting, November 16
 Regression Analysis, October 18, December 14
 Report Formatting, October 25, December 28
 Saving Money with N- and B-tasks, October 4
 Text Editing, November 19-21

SEMINARS/SPECIAL COURSE (continued)

Vienna (All courses are in German unless marked *)

- APL 1 fuer EDV-Leute, October 3-4
- Innere und Aussere Produkte in SHARP APL, October 10
- Batch Tasks in APL, October 24
- APL 2, November 6-8
- APL 1 for EDP People *, November 14-15
- Packages als Datentyp, November 21
- MABRA: Ein Management-Informationssystem, Nov. 28
- Event Trapping, December 5
- APL 1, December 11-13
- Die $\alpha\omega$ -notation, December 19

Warrington

- Appreciation of APL, December 5

Washington DC

- Data Base Design, December 14 (1 day)
- MAGIC for Time Series Anal., Nov. 12-13 (full days)
- Statistical Pkges in SHARP APL, October 18 (half day)
- SHARP APL Enhancements, September 27-28 (half days)

INTERMEDIATE APL

- Duesseldorf, Oct. 4,5 (+ Intro.)
- Gloucester, Dec. 3
- London, Oct. 23/Nov. 22/Dec. 13
- Montreal, Nov. 27-29 (Eng./Fr.)
- Ottawa, Jan, 22,23
- San Francisco, Oct. 9, Dec. 4
- Toronto, Nov. 12-14
- Warrington, Oct. 17

INTRODUCTION TO APL

Atlanta

- (5 day)
- September 24-28

Birmingham

- (3 day)
- October 16-18
- December 11-13

Duesseldorf

- (Combination)
- Intro: Oct. 1-3
- Interm: Oct. 4-5

Gloucester

- (3 days)
- October 1-3

London

- (3 day)
- October 10-12
- November 5-7
- December 5-7

Montreal

- (English)
- September 11-13
- November 13-15
- (French)
- October 2-4
- December 11-13

Ottawa

- (non-programmers)
- October 2-4
- December 4-6
- February 5-7
- (programmers)
- November 6-8
- January 8-10

Rochester

- September 24-28
- October 15-19
- November 19-23

San Francisco

- (4 half days)
- October 2-5
- November 13-16

Toronto

- (3 day)
- September 19-21
- October 1-3
- November 5-7
- December 3-5
- December 17-19

Vancouver

- September 19-21
- October 24-26
- November 21-23
- January 23-25
- February 20-22

Vienna

- (see Special Courses)
- October 3-4
- November 14-15
- December 11-13

Warrington

- (3 day)
- December 11-13

Washington

- (half days)
- September 17-20
- October 29-Nov. 1

Zurich

- (APL Einfuehrung 1)
- September 26-28
- October 15-17
- November 12-14
- December 3-5
- (Wiederholung Einf. 1)
- October 1
- October 29
- November 19
- December 10
- (APL Einf. 2)
- October 2-3
- October 30-31
- November 20-21
- December 11-12



International Branch Offices

Aberdeen

I.P. Sharp Associates Limited
5 Bon Accord Crescent
Aberdeen AB 12DH
Scotland
(0224) 25298

Amsterdam

Intersystems B.V.
Herengracht 244
1016 BT Amsterdam
The Netherlands
(020) 24 40 50
Telex: 18795 ITS NL

Atlanta

I.P. Sharp Associates, Inc.
5000 Snapfinger Woods Dr.
Decatur, Georgia 30035
(404) 987-2301

Boston

I.P. Sharp Associates, Inc.
Suite 415
148 State Street
Boston, Massachusetts 02109
(617) 523-2506

Brussels

I.P. Sharp Europe S.A.
Avenue General de Gaulle, 39
1050 Bruxelles
Belgique
(02) 649 99 77

Calgary

I.P. Sharp Associates Limited
Suite 2660, Scotia Centre
700-2nd Street S.W.
Calgary, Alberta T2P 2W2
(403) 265-7730

Chicago

I.P. Sharp Associates, Inc.
2 North Riverside Plaza
Suite 1736
Chicago, Illinois 60606
(312) 648-1730

Cleveland

I.P. Sharp Associates, Inc.
(216) 431-6861
(local call, switched through
to Rochester office.)

Copenhagen

I.P. Sharp ApS
Østergade 24B
1100 Copenhagen K
Denmark
(01) 112 434

Coventry

I.P. Sharp Associates Limited
7th Floor B Block
Coventry Point, Market Way
Coventry, England CV1 1EA
(0203) 21486/7

Dallas

I.P. Sharp Associates, Inc.
Suite 1148, Campbell Centre
8350 Northcentral Expressway
Dallas, Texas 75206
(214) 369-1131

Dublin

Gamma Data Systems Limited
(Agent)
Dollard House
Wellington Quay
Dublin 2, Ireland
(0001) 711877

Düsseldorf

I.P. Sharp GmbH
Leostrasse 62A
4000 Düsseldorf 11
West Germany
(0211) 57 50 16

Edmonton

I.P. Sharp Associates Limited
Suite 505
10065 Jasper Avenue
Edmonton, Alberta T5J 3B1
(403) 428-6744

Gloucester

I.P. Sharp Associates Limited
29 Northgate Street
Gloucester, England
(0452) 28106

Houston

I.P. Sharp Associates, Inc.
Suite 925, One Corporate Square
2600 Southwest Freeway
Houston, Texas 77098
(713) 526-5275

London, Canada

I.P. Sharp Associates Limited
Suite 510, 220 Dundas Street
London, Ontario N6A 1H3
(519) 434-2426

London, England

I.P. Sharp Associates Limited
132 Buckingham Palace Road
London SW1W 9SA
England
(01) 730-0361

Melbourne

I.P. Sharp Associates Pty. Ltd.
36 Elizabeth Street
South Yarra
Victoria, Australia 3141
(03) 244-417

Miami

I.P. Sharp Associates, Inc.
Suite D, Kennedy Building
14560 N.W. 60th Avenue
Miami Lakes, Florida 33014
(305) 556-0577

Milan

I.P. Sharp Srl
Via Eustacchi, 11
20129 Milan
Italy
(2) 271 6541/221 612

Minneapolis

I.P. Sharp Associates, Inc.
Mail to Chicago
Suite 1371, 1 Appletree Square
Bloomington, Minnesota 55420
(612) 854-3405

Montreal

I.P. Sharp Associates Limited
Suite 1610
555 Dorchester Boulevard W.
Montreal, Quebec H2Z 1B1
(514) 866-4981

New York City

I.P. Sharp Associates, Inc.
Mail to:
Suite 2004
200 Park Avenue
New York, N.Y. 10017
(212) 986-3366
Also at:
Suite 242, East. Mezz.
(212) 599-0232

Newport Beach

I.P. Sharp Associates, Inc.
Suite 1135
610 Newport Center Drive
Newport Beach, Ca. 92660
(714) 644-5112

Oslo

I.P. Sharp A/S
Kronprinsesse Märthas Plass 1
Mail to:
Postboks 1470, Vika
Oslo 1, Norway
(02) 41 17 88

Ottawa

I.P. Sharp Associates Limited
Suite 600, 265 Carling Ave.
Ottawa, Ontario K1S 2E1
(613) 236-9942

Palo Alto

I.P. Sharp Associates, Inc.
Suite 201, 220 California Ave.
Palo Alto, Ca. 94306
(415) 327-1700

Paris

Société I.P. Sharp SARL
Tour Neptune - Cedex No.20
92086 Paris-la-defense
France
(1) 773 57 77

Philadelphia

I.P. Sharp Associates, Inc.
Suite 407, 1420 Walnut Street
Philadelphia, PA. 19102
(215) 735-3327

Phoenix

I.P. Sharp Associates, Inc.
3033 N. Central Avenue
Phoenix, Arizona 85012
(602) 264-6819

Rochester

I.P. Sharp Associates, Inc.
1200 First Federal Plaza
Rochester, N.Y. 14614
(716) 546-7270

San Francisco

I.P. Sharp Associates, Inc.
Suite C-415, 900 North Point St.
San Francisco, Ca. 94109
(415) 673-4930

Saskatoon

I.P. Sharp Associates Limited
Suite 208, 135 21st Street E.
Saskatoon, Sask. S7K 0B4
(306) 664-4480

Seattle

I.P. Sharp Associates, Inc.
Suite 217, Executive Plaza East
12835 Bellevue-Redmond Road
Bellevue, Washington 98005
(206) 453-1661

Stockholm

I.P. Sharp AB
Kungsgatan 65
S111 22 Stockholm, Sweden
(08) 21 10 19

Stuttgart/Boeblingen

I.P. Sharp GmbH
Schafgasse 3
7030 Boeblingen
West Germany
(070 31) 2 30 14

Sydney

I.P. Sharp Associates Pty. Ltd.
Suite 1342, 1175 Pitt Street
Sydney, N.S.W., Australia 2000
(02) 232-5914

Toronto

I.P. Sharp Associates Limited
145 King Street West
Toronto, Ontario M5H 1J8
(416) 364-5361

Vancouver

I.P. Sharp Associates Limited
Suite 604, 1112 West Pender St.
Vancouver, B.C. V6E 2S1
(604) 682-7158

Victoria

I.P. Sharp Associates Limited
Chancery Court
1218 Langley Street
Victoria, B.C. V8W 1W2
(604) 388-6365

Vienna

I.P. Sharp Ges.mBH
Rechte Wienzeile 5/3
1040 Wien, Austria
(222) 57 65 71

Warrington

I.P. Sharp Associates Limited
Paul House
89-91 Buttermarket Street
Warrington, Cheshire
England WA1 2NL
(0925) 50413/4

Washington

I.P. Sharp Associates, Inc.
Suite 400, 1835 K Street N.W.
Washington, D.C. 20006
(202) 293-2915

Winnipeg

I.P. Sharp Associates Limited
Suite 208
213 Notre Dame Avenue
Winnipeg, Manitoba R3B 1N3
(204) 947-1241

Zurich

I.P. Sharp A.G.
Badenerstrasse 141
8004 Zurich
Switzerland
(1) 241 52 42

SHARP APL Communications Network: Local Access Cities

APL OPERATOR VOICE (416) 363-2051 COMMUNICATIONS (416) 363-1832

Local dial access is available in all locations listed above. The SHARP APL Communications Network also provides local dial access in:

- Ann Arbor • Buffalo • Coventry • Dayton • Des Moines • Detroit • Ft. Lauderdale • Greene (NY) • Greenwich (Ct)
- Halifax • Hamilton • Kitchener • Liverpool • Los Angeles • Manchester • Raleigh • Regina • Syracuse • White Plains (NY).

In the United States the SHARP APL Network is interconnected with the networks of TYMNET and TELENET to provide local dial access in more than 100 other cities.