Artificial Intelligence Project--RLE and MIT Computation Center
Memo 20--Puzzle Solving Program in LISP

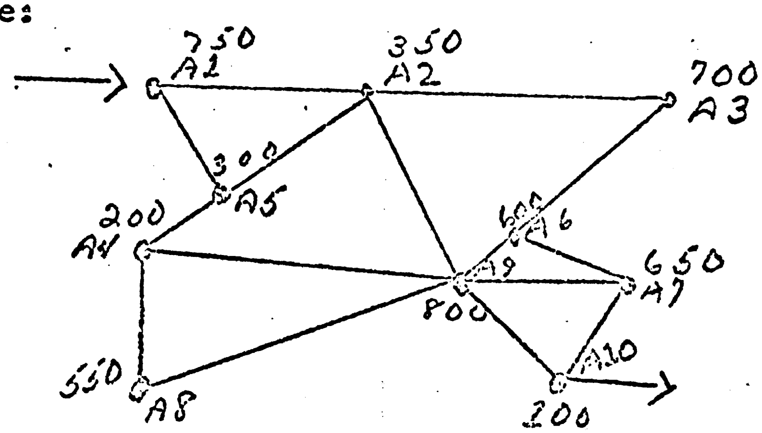
by

John McCarthy

In this note we give as an example of LISP programming a function for solving a class of puzzles in a recent prize contest.

A linear graph is given with a starting point and an endpoint. Associated with each vertex of the graph is a numerical value. The object is to find a path on the graph from the starting point to the endpoint which goes through no vertex twice and which maximizes the sum of the values associated with the vertices through which the path goes.

Example:



We shall use the following conventions in our LISP program for solving the problem:

1. The graph is described by assigning to each vertex an atomic symbol and giving a list of terms one for each vertex, each of which is a list of three items: the name of the vertex, a list of the vertices immediately accessible from it, and the value assigned to the vertex. The graph in the example is described by:

((A1,(A2,A5),7.5),(A2,(A1,A5,A9,A3),3.5),(A3,(A2,A6),7.0),(A4,(A5,A9,A3),2.0)),(A5,(A1,A4,A2),3.0),(A6,(A3,A9,A7),6.0),(A7,(A6,A9,A10),6.5),(A8,(A4,A9),5.5),(A9,(A8,A4,A2,A6,A7,A10),8.0),(A10,(A7,A9,H),1.0))

2. A position is described by list of the vertices covered up to that point in reverse order. The initial position is (A1) in the example on the previous page.

3. The list of squares to which it is legal to move in a position p is given by

legals[p] = less [car][ass2[car[p];pz1]];p]

where

less[m;n] = [null[m] → NIL;occur[car[m];n]→ less[cdr[m];n];T → cons[car[m];less[cdr[m];n]]]

and

ass2[x;m] = [eq[caar[m];x] → cadr[m];T → ass2[x;cdr[m]]]

and

occur[x;n] = ~null[n] ∧ [eq[x;car[n]] ∨ occur[x;cdr[n]]]

Here less [m;n] is a list of those elements of m not members of n. ass2[x;m] is cdr [the first element of m, car of which is x]. occur [x;m] is the predicate that asserts that x is a member of m.

4. The best path is given by best[initial position; moves from the initial position; path with very low value] or in the present example, best[(A1);(A2,A5);(A1)]. The function best [p;m;s] is the best path starting with the subpath p, where the moves from p are to be taken from the list m and s is the best path found so far. We have

best[p;m;s] = [null[m] → s;T → best[p;cdr[m]; λ[[n];[eq[car[n]; H] → better[p;s] → p;T → s]; T → best[n;legals[n];s]]][cons[car[m]; p]]]]

For comparing two paths we have

better[p1;p2] = eq[MINUS;car[sum[addup[p2]; list[MINUS; addup[p1]]]]]

where

addup [p] = [null[p] → 0.0; T → sum[value [car[p]];addup [cdr[p]]]]

and

value[sq] = cadr[ass2[sq;pz1]]

CS-TR Scanning Project
Document Control Form

Date : 11 / 30 / 95

Report # AIM - 20

Each of the following should be identified by a checkmark:
Originating Department:

- [x] Artificial Intellegence Laboratory (AI)
- [ ] Laboratory for Computer Science (LCS)

Document Type:

- [ ] Technical Report (TR)
- [x] Technical Memo (TM)
- [ ] Other: ____________

Document Information

Number of pages: 2 (6 - IMAGES)
Not to include DOD forms, printer intstructions, etc... original pages only.

Originals are:
- [x] Single-sided or
- [ ] Double-sided

Intended to be printed as :
- [x] Single-sided or
- [ ] Double-sided

Print type:
- [ ] Typewriter
- [ ] Offset Press
- [ ] Laser Print
- [ ] InkJet Printer
- [ ] Unknown
- [x] Other: COPY OF MIMEOGRAPH

Check each if included with document:
- [ ] DOD Form
- [ ] Funding Agent Form
- [ ] Cover Page
- [ ] Spine
- [ ] Printers Notes
- [ ] Photo negatives
- [ ] Other: ____________

Page Data:

Blank Pages(by page number): ____________

Photographs/Tonal Material (by page number): ____________

Other (note description/page number):

Description :	Page Number:
IMAGE MAP! (1-2)	1-2
	(3-6) SCANCONTROL, TRGT'S (3)

Scanning Agent Signoff:
Date Received: 11 / 30 / 95 Date Scanned: 12 / 12 / 95 Date Returned: 12 / 13 / 95

Scanning Agent Signature: Michael W. Cook

Rev 9/94 DS/LCS Document Control Form cstrform.vsd