```
      mark chosen path;
      reduce T, 𝒮, and the labels in L' by the attributes
            chosen path label;
      remove any empty labels from L' created by the
            reduction
    end
    end;
    if l ≠ ∅ then
    begin
      L'' := L'' ∪ {l};
      L' := L' − {l}
    end
    end;
    while 𝒮 ≠ ∅ do
    begin {select any remaining query attributes that have
          not been added to the join sequence}
      choose l ∈ L'';
      choose lowest frequency attribute added at the
            terminal step;
      choose path contributing the most new 𝒮 attributes,
            in case of tie use the shortest path;
      mark the chosen path;
      reduce 𝒮 and the labels in L'' by the attributes in
            the chosen path label;
      remove any empty labels from L'' created by the
            reduction
    end;
      path-length := number of edges in marked paths
  end;
```

Algorithm 3.2. Creation of the target query hypergraph (Q_T)

```
begin
  R := ∅; R' := ∅;
  let 𝒮 := vertices in the source query hypergraph;
  let J := nodes in the join sequence;
  assign the first element of J to R;
  let J := J − R; weight := 1;
while 𝒮 ⊈ R do
begin
  assign the first element of J to R'; {note that a node in
        I_R represents an edge in Q_T}
  generate a join edge for the attributes in R ∪ R' labelled
        with j_weight;
  weight := weight + 1;
  let R := R ∪ R'; J := J − R'
end;
  copy labels from the vertices in Q_S to the vertices in
        R;
  insert the "OR" edges in R from Q_S;
  insert the projection edge in R from Q_S
end.
```

Algorithm 3.3. Mapping target query hypergraph to relational algebra

```
begin
  i := 1; {join edge weight}
  F := ∅; {selection condition}
  J := ∅; {set of joined relation schemes}
  P := the set of vertices in the project edge;
if join edges exist
then
begin
  choose join edge j_i; {use first join edge j_1}
  let J := the join of the system edges in j_i; {perform
        joins}
if any vertices in j_i are labelled {create selection formula
      for j_1}
then
  let F := the AND of all labels within j_i;
  i := i + 1 {point to next join edge}
end;
while a join edge j_i exists do
begin
  let J := the join of the system edges in j_i; {perform
        joins}
if any vertices in j_i − j_{i−1} are labelled {expand selection
formula}
then
begin
  let condition be the AND of all labels in j_i − j_{i−1};
  F := F AND condition
end;
  i := i + 1 {point to next join edge}
end;
  generate 'Π_P(σ_F(J))'
end.
```


Announcement

1–5 August 1988

10th Congress of the International Ergonomics Association, Sydney, Australia

The 10th International Ergonomics Congress, to be held in Sydney from 1 to 5 August 1988, has released its provisional programme and registration details. An impressive 32-page booklet, it contains details on registration, the scientific programme, keynote speakers, social and accompanying persons' programmes, and associated meetings.

'Designing a Better World' is the challenging congress theme, and an imaginative programme reflects this challenge.

Two post-Congress tours are offered: one to Australia's 'Red Centre' and famous Kakadu National Park in 'Crocodile Dundee' territory, the other to the beautiful Great Barrier Reef in Northern Queensland. A weekend escape on the waterways close to Sydney is offered for those with limited time to take a break.

The programme also contains general information about Australia, travel and accommodation details.

Enquiries to:
IEA88 Secretariat, PO Box 380, Spit Junction, NSW 2088, Australia. Tel. 61 2 969 1400. Fax 61 2 908 4982.