## Goto algorithm

Implement the Goto algorithm. Take advantage of the provided starter code, so that you can use existing functions for allocation, initialization, etc. Consult the lecture notes and the instructors if you have any questions. You may use this diagram as a reference.

Lab II

Starter project

## Warmup – conditional sum

Create a vector of random floating point numbers from the interval (0, pi/2). Compute the sum of the numbers whose sine is larger than 0.5. Measure the throughput of your approach and pass the vector size as an argument to your benchmark. Now sort the values before computing the sum. How does this impact execution time? Create 2 benchmarks – one which includes the sort in the measurement, and one which doesn't.

LAB 2

When is sorting a viable strategy?

## Matrix-vector product

Implement a matrix-vector product. Consider the following strategies for matrix storage:

- Vector of M vectors of size N
- Flat one big vector of size M\*N (index appropriately)

Where M, N are the matrix dimensions. How does loop interchange affect the solution time?

Write 4 benchmarks parametrized by M and N (or assume M == N for simplicity). Discuss the results.