

## 6.45 ♦♦♦♦

In this assignment, you will apply the concepts you learned in Chapters 5 and 6 to the problem of optimizing code for a memory-intensive application. Consider a procedure to copy and transpose the elements of an  $N \times N$  matrix of type `int`. That is, for source matrix  $S$  and destination matrix  $D$ , we want to copy each element  $s_{i,j}$  to  $d_{j,i}$ . This code can be written with a simple loop,

```
1  void transpose(int *dst, int *src, int dim)
2  {
3      int i, j;
4
5      for (i = 0; i < dim; i++)
6          for (j = 0; j < dim; j++)
7              dst[j*dim + i] = src[i*dim + j];
8  }
```

where the arguments to the procedure are pointers to the destination (`dst`) and source (`src`) matrices, as well as the matrix size  $N$  (`dim`). Your job is to devise a transpose routine that runs as fast as possible.