Notes on Loop Coalescing
CS4961
September 27, 2011

This note is just a short example on loop coalescing to help you with your homework.

Loop coalescing involves reducing the number of loops in a nest by linearizing the multi-dimensional loop nest into a single dimension. A coalesced loop might achieve higher performance on certain architectures (but usually will not pay off in modern architectures). As I mentioned in class, I really assigned this to get you to think about linearizing multi-dimensional spaces, and what it means.

I'll first introduce a simple example, and then the example we worked through on the board. Suppose I start with:

```
for (i=0; i<N; i++)
    for (j=0; j<M; j++)
        a[i][j] = b[j][i] + 1;
```

After coalescing, this will become:

```
for (ij=0; ij<N*M; ij++) {
    i = ij/M;
    j = ij % M;
   a[i][j] = b[j][i] + 1;
}
```

The example from the board adds a little complexity by using a lower bound that is not 0. Here is the original code:

```
for (i=lb_i; i<ub_i; i++)
   for (j=lb_j; j<ub_j; j++)
       a[i][j] = b[j][i] + 1;
```

And the coalesced code:

```
iters_i = ub_i – lb_i + 1;
iters_j = ub_j – lb_j + 1;
for (ij=0; ij<iters_i*iters_j; ij++) {
    i = (ij/iters_j) + lb_i;
    j = ij % iters_j + lb_j;
   a[i][j] = b[j][i] + 1;
}
```

Since the second example is tricky, you should probably use the initial example to help you reason about your homework.