IN3200/IN4200 Exercise Set 5

Exercise 1

Calculate code balance for the following loops, assuming that all arrays have to be loaded from memory. (Arrays `A`, `B` and `y` contain double-precision values.)

(a) Dense matrix-vector multiplication:

```
for (j=0; j<N; j++)
  for (i=0; i<N; i++)
    y[j] += A[j][i]*B[i];
```

(b) Vector norm:

```
double s = 0.;
for (i=0; i<N; i++)
  s += A[i]*A[i];
```

(c) Scalar product:

```
double s = 0.;
for (i=0; i<N; i++)
  s += A[i]*B[i];
```

(d) Scalar product with indirect access:

```
double s = 0.;
for (i=0; i<N; i++)
  s += A[i]*B[K[i]];
```

Assume that each entry `K[i]` is a random integer between 0 and $N-1$.

Exercise 2

Please apply "unroll and jam" (with 4 as the unroll depth) to the outer loop of the following code:

```
for (j=0; j<N; j++)
  for (i=0; i<=j; i++)
    y[j] += A[j][i]*B[i];
```

Note: The value of `N` may not be divisible by 4.

Exercise 3

Please optmize the following function:

```
void foo (int N, double **mat, double **s, int *v) {
  int i,j;
  double val;

  for (j=0; j<N; j++)
    for (i=0; i<N; i++) {
      val = 1.0*(v[j]%256);
      mat[i][j] = s[i][j]*(sin(val)*sin(val)-cos(val)*cos(val));
    }
}
```

Assume that the above function will be repeatedly called, with changing contents of the input arrays `s` and `v`.