CS152 Section 9

Q1: Vector Lane Trade-off

Consider the following vector instruction sequence:

```
vadd v1, v2, v3
vadd v4, v5, v6
```

The next instruction after that uses a different functional unit. Suppose that VL=32.   On which machine would this code perform better: a design with 8 lanes and 2 cycles of dead time or a design with 16 lanes and 8 cycles of dead time? Assume a single-cycle ALU.

Q2: Vectorization

Vectorize the following code:

```
for (i = 0; i < M; i++) {
    C[i] = A[2*i+1] + A[2*i] * B[2*i];
}
```

x1 holds a pointer to array A, x2 holds a pointer to array B, x3 holds a pointer to array C, and x4 holds M. The array elements are single-precision values. Assume that the arrays do not overlap in memory. You should not assume that M is an integer multiple of the maximum vector length.

Q3: Vectorization

How might the following code be vectorized? Clearly state any assumptions that you used for your answer for what the architecture provides, such as specific instructions, registers, etc.

```
for (i = 0; i < N; i++) {
    if (A[i+1])
        A[i] = A[i] + B[C[i]];
}
```