1. Partitioning the following C code to a loosely-coupled coprocessor design. Using the profile information annotated within C code, partition as many loops as possible to hardware coprocessors to increase performance. Estimate the speedup of the partitioned design over software only execution.

**Extra Credit:** The student(s) will the greatest **verifiable** speedup will receive 10 points extra credit.

```c
int main() // Total Cycles: 8193437
{
    int n;
    int i, j, k;
    for (n = 0 ; n < LOOPS ; n++) // Total Cycles: 8186006, Execs: 1, Iters: 1000
        {
            for(i=1;i<=SIZE;i++)       // Total Cycles: 579000, Execs: 1000, Iters: 5
                for(j=1;j<=SIZE;j++)    // Total Cycles: 520000, Execs: 5000, Iters: 5
                    c[i][j] = 0;
            for(i=1;i<=SIZE;i++)       // Total Cycles: 7579000, Execs: 1000, Iters: 5
                for(j=1;j<=SIZE;j++)    // Total Cycles: 7520000, Execs: 5000, Iters: 5
                    for(k=1;k<=SIZE;k++) // Total Cycles: 7225000, Execs: 25000, Iters: 5
                        c[i][j] += a[i][k] * b[k][j];
        }
    return 0;
}
```