For the following code, identify and describe the race condition. How would you protect this race condition if parallelizing the loop in OpenMP

int main(int argc, char \*\* argv) {

int i;

int sum = 0;

for(i=0;i<100;i++) {

sum += i;

}

printf("sum = %d\n",sum);

}

For the following code, identify and describe the loop carried dependency.

#define N 10

int main(int argc, char \*\* argv) {

int i;

int fib[N];

fib[0] = 1;

fib[1] = 2;

for(i=2;i<N;i++) {

fib[i]=fib[i-1]+fib[i-2];

}

printf("fib[N-1] = %d\n",fib[N-1]);

}

The Ensemble Based Simulated Annealing (EBSA) Algorithm is modified to avoid a loop carried dependency by replacing one annealer taking many steps with many annealers taking fewer steps. Describe how this makes EBSA a different algorithm from SA.