1.
What is the problem with the following code?

```
struct T {
    int a;
    size_t b;
};
T array[arraySize];

size_t i;
#pragma omp parallel sections num_threads(2)
{
   #pragma omp section
   {
      for (i = 0; i != arraySize; ++i)
         array[i].a = 1;
   }
   #pragma omp section
   {
      for (i = 0; i != arraySize; ++i)
         array[i].b = 2;
   }
}
```

2.
Use OpenMP to parallelize the following code. What would happen if this was a one-dimensional array, in a single for loop, and the same parallelization was used?

```
int i, j;

for(i=0; i<n; i++)
      for(j=1; j<n; j++) {
            array[i][j]+= array[i][j-1];
      }
```

1D case:

```
      for(j=1; j<n; j++)
            array[j]+= array[j-1];
```

3.
Optimize the following code, using OpenMP.

```
void hello(long *old, long *new, int n) {
    int i;
    double sumWeights=0, sum=0;

    for(i = 0; i < n; i++) {
        new[i] = old[i] * exp(100.0f/old[i]);
        sum += old[0];
    }

    for(i = 0; i < n; i++)
        sumWeights += new[i]/sum;

    for(i = 0; i < n; i++)
        new[i] = new[i]/sumWeights;
}
```

4.
What are the differences between dynamic and static linking? What are some advantages and disadvantages?

5.
What type of exception would each of the following lead to? Are they synchronous or asynchronous exceptions? What is their return behavior?

a. Dividing by 0

b. Tired of waiting for your “optimized” code for the OpenMP lab, you terminate your process by pressing Ctrl-C at the keyboard

c. The MMU fetches a PTE from the page table in memory, but the valid bit is zero

d. You create a file using the open() system call

6. (Textbook 9.3)
Given a 32 bit virtual address space and a 24-bit physical address, determine the number of bits in the VPN, VPO, PPN, and PPO for the following page sizes P:

P	VPN bits	VPO bits	PPN bits	PPO bits
1 KB				
2 KB				
4 KB				
8 KB				