Exercises for Week 9
Textbook Sections 5.1 to 5.3

Question 1

Consider the following C function for vector dot product:

```
double dot_product (double A[], double B[], int size) {
  int i;
  double p = 0.0;
  for (i = 0; i < size; i++)
    p = p + A[i] * B[i];
  return p;
}
```

a. Which of the variables (if any) exhibit temporal locality? Why?

b. Which of the variables (if any) exhibit spatial locality? Why?

c. Which of the variables (if any) exhibit no or poor locality? Why?

Question 2

Using the data from Figure 5.5, extrapolate the trend to estimate the row access time and column access time for the 2015 and 2018 generations of DRAMs. Are you able to find any data on DRAMs for the 2015 generation to compare with your extrapolation?

Question 3

Consider a direct-mapped data cache of size 64Kbytes with block size of 16 bytes.

a. Show how a 32-bit data address is subdivided into tag, index and offset fields, giving the number of bits in each field.

b. How many bits of storage are required for the cache in addition to the 64Kbytes of data storage?

For the dot product function in Question 1, suppose the vectors A and B are 100 elements each. Assume the variables size, i and p are all allocated in registers.

c. What would be the best-case miss rate in the cache? Why?

d. What would be the worst case miss rate? Why?

Question 4

Textbook Exercise 5.5 <§5.3>

For a direct-mapped cache design with a 64-bit address, the following bits of the address are used to access the cache.

- Tag: bits 63:10
- Index: bits 9:5
- Offset: bits 4:0

a. What is the cache block size (in words)?

b. How many entries does the cache have?

c. What is the ratio between total bits required for such a cache implementation over the data storage bits?

Beginning from power on, the following byte-addressed cache references are recorded.

Hex	000	004	010	084	0E8	0A0	400	01E	08C	C1C	0B4	884
Dec	0	4	16	132	232	160	1024	30	140	3100	180	2180

d. For each reference, list (1) its tag, index, and offset, (2) whether it is a hit or a miss, and (3) which bytes were replaced (if any).

e. What is the hit ratio?

f. List the final state of the cache, with each valid entry represented as a record of <index, tag, data>.