1. What does encapsulation mean? How does encapsulation help control the complexity of multiprogrammer applications?
2. Identify the scope of each variable in the following:

vector<int> v;

namespace Acme\_Search

{

const int maxdata = 100;

int binary\_search(int from, int to, int a)

{

if (from > to)

return -1;

int mid = (from + to) / 2;

if (v[mid] == a)

return mid;

else if (v[mid] < a)

return binary\_search(mid + 1, to, a);

else

return binary\_search(from, mid - 1, a);

}

}

1. What does it mean for one variable to shadow another?
2. How is protected visibility different from private or public visibility?
3. What does friendship mean? What are the positive benefits and negative consequences of friendship?

1. In what ways is a nested class similar to a friend class? In what ways is it different?
2. How is private inheritance different from public inheritance?
3. In what situation is private inheritance preferable to public inheritance?
4. Explain in which sense classes are closed while name spaces are open.
5. When would you define a class as a nested class, and when would you define it in a name space?
6. Suppose Harry J. Hacker develops a code library that he wants others to use. Why would it not be a good idea to place it into a name space hjh? What name space name might be appropriate?
7. Why is it acceptable to use short aliases for name spaces, even though short names for name spaces are not appropriate?