

1. Assume that each of the following statement applies to the same program.
  - a) Write a statement that opens file `transaction.dat` for input; use an `ifstream` object called `inTransaction`.
  - b) Write a statement that opens file `balance.dat` for input; use an `ifstream` object called `inBalance`.
  - c) Write a statement that reads integer `accountNumber`, floating-point `amount` from the file `transaction.dat`; use `ifstream` object `inTransaction`.
  - d) Write a statement that reads integer `accountNumber`, string `name`, floating-point `currentBalance` from the file `balance.dat`; use `ifstream` object `inBalance`.
  - e) Write a brief code snippet that writes integer `accountNumber`, string `name`, updated balance which is a floating-point `currentBalance + amount` to the file `balance.dat`.

```
a) ifstream inTransaction("transaction.dat");
b) ifstream inBalance("balance.dat");
c) inTransaction >> accountNumber >> amount;
d) inBalance >> accountNumber >> name >> currentBalance;
e) ofstream outBalance("balance.dat"); outBalance << accountNumber << name <<
    currentBalance + amount;
```

2. Suppose that `numStudents` is an `int` variable and `classCode` is a `string` variable. What are the values of `numStudents` and `classCode` after the following input statements execute:

```
cin >> numStudents;
getline(cin, classCode);
```

if the input is:

- (a) 80 ENGG1111
- (b) 80  
ENGG1111

```
a) numStudents = 80, classCode = " ENGG1111"
b) numStudents = 80, classCode = ""
```

3. The following program is supposed to read two numbers from a file named `numbers.dat` and write the product of the numbers to a file named `product.dat`. However, it fails to accomplish the task. Fix by rewriting the program so that it performs what it is supposed to do.

```
#include <iostream>
#include <fstream>
using namespace std;

int main()
{
    int num1, num2;
    ifstream infile;

    outfile.open("product.dat");
    infile >> num1 >> num2;
    outfile << "Product = " << num1 * num2 << endl;
    return 0;
}
```

```

#include <iostream>
#include <fstream>
using namespace std;

int main()
{
    int num1, num2;
    ifstream infile;
    ofstream outfile;

    infile.open("numbers.dat");
    outfile.open("product.dat");

    infile >> num1 >> num2;
    outfile << "Product = " << num1 * num2 << endl;
    infile.close();
    outfile.close();

    return 0;
}

```

4. Consider the following statements:

```

struct movieType
{
    string title;
    string genre;
    int year;
    double rating;
};
movieType    movies[100];
movieType    oldMovie;

```

State if each of the following statements is valid or invalid. If a statement is invalid, explain why.

- (a) `cout << oldMovie.name;`
- (b) `movies.year = 2015;`
- (c) `movies[11] = oldMovie;`
- (d) `oldMovie.title = "Titanic";`
- (e) `if (movies[99].genre == "drama")`  
`movies[99].rating = 3.5;`

- (a) invalid, there is no member named "name" in the structure movieType
- (b) invalid, movies is an array, not a variable of type movieType
- (c) valid
- (d) valid
- (e) valid

5. The following program calculates the summation of first  $n$  natural numbers. E.g., if  $n = 6$ ,  $sum = 1 + 2 + 3 + 4 + 5 + 6 = 21$ . Rewrite the `sum()` that uses recursion to calculate and return the sum of first  $n$ .

```
#include <iostream>
using namespace std;

/*
// iterative version
int sum(int n)
{
    int sum = 0;
    for (int i = 1; i <= n; ++i)
        sum += i;

    return sum;
}
*/

// recursive version
int sum(int n)
{
    if (n == 1)
        return 1;
    else
        return n + sum(n-1);
}

int main()
{
    int n;

    cout << "Enter a positive integer: ";
    cin >> n;

    cout << "Sum of first " << n << " natural numbers = " << sum(n) << endl;

    return 0;
}
```