

Name: _____

SSN: _____

Login: _____

CS 12 - Final March 25, 1998

Be sure to read each problem carefully and follow the directions. Points may be marked off if you do not follow the directions. For example, if the problem asks you to write a function, do not write an entire program. Please feel free to ask if you have any questions.

Problem 1	24	
Problem 2	16	
Problem 3	12	
Problem 4	10	
Problem 5	12	
Problem 6	10	
Problem 7	6	
Problem 8	10	
Problem 9	15	
Problem 10	10	
Problem 11	60	
TOTAL	185	

1. (24 pts) True/False (please write out “True” and “False”, do not just put a T or F).

- (a) Input using the getline function reads over all leading whitespace.
- (b) Each statement that processes a file in a C++ program explicitly refers to that file by name.
- (c) Using the iomanip library, one can print out a number in hexadecimal and binary bases.

(d) The code:

```
char *s1 = ‘rain’;  
char s2[10] = ‘bow’;  
char s3[10];
```

```
cout << strcat( s1, strcpy(s3, s2));
```

will print “rainbow”.

- (e) To the class in which they are written, there is no difference if a member has been declared to be protected or private.
- (f) Anything that can be solved with operator overloading can be solved with a regular function.
- (g) Destructors to all objects are called at the end of the program.
- (h) A side affect of operator overloading is that the precedence of that operator may be changed.
- (i) Constructors and destructors can be overloaded.
- (j) Writing a function in a derived class that overloads a function in the base class will make the base class function inaccessible.
- (k) If class A is a friend of class B, then class B is a friend of class A.
- (l) The following two functions can be overloaded.

```
int func1(int x, char c){}  
void func1(int y, char ch){}
```

2. (16 pts) Fill in the blank.

- (a) Overloaded stream operators are often defined as _____ function of a class.
- (b) Input operations are supported by the _____ class.
- (c) To find if a specific character exists within a string, the _____ function is used.
- (d) Getting space for an object at run-time is called _____.
- (e) _____ should be used when you have a class that is the subset of another class that needs to be defined.
- (f) When the insertion operator is overloaded, the return type is _____.
- (g) A _____ function must always have a base case in order to avoid an infinite loop.
- (h) _____ functions decide which function to call at run-time.

3. (12 pts) Define the following terms.

(a) virtual function

(b) constructor

(c) template function

4. (10 pts) Given the following code:

```
class book{
    private:
        char title[40];
        char author[30];
        int num_pages;
};

book math;
double nums[20][20];
double *ptr1 = nums[0][0];
book library[10];
book *ptr2 = &math;
```

What are the types of the following expressions? Write in the letter of the corresponding correct answer. An answer may be used more than once.

- | | | | |
|--------|-------------------|---|---------------------|
| ___ 1 | math | A | integer |
| ___ 2 | nums[0] | B | double |
| ___ 3 | library[3].title | C | character |
| ___ 4 | ptr1 | D | class |
| ___ 5 | book | E | double pointer |
| ___ 6 | author | F | integer pointer |
| ___ 7 | ptr2->num_pages | G | string |
| ___ 8 | *ptr2 | H | array of characters |
| ___ 9 | book.title | I | array of doubles |
| ___ 10 | library.num_pages | J | book |
| | | K | invalid |
| | | L | none of the above |

5. (12 pts) Show the values in the array **A** after the following code has been executed.

```
int A[6] = {1,2,3,4,5,6};
int *ip = A;
*ip = 16;
ip++;
(*ip)++;
ip++;
*ip = *(ip + 2);
ip += 2;
*ip += 10;
*(ip + 1) = 20;
```

6. (10 pts) Show the output of the following code. Be specific about all characters.

```
char buf[40];
char ch;

infile >> buf;
cout << buf << endl;

infile.get(ch);
cout << ch << endl;

infile.getline(buf, 40, '\n');
cout << buf << endl;

infile >> buf;
cout << buf << endl;

infile.getline(buf, 40, '\n');
cout << buf << endl.
```

Assume that the file being read from contains the following:

```
Have a nice
spring
break
```


9. (15 pts) A file called *names.dat* contains integers associated with people. For example:

```
100 Alan Greenspan
421 Paula Jones
206 Madonna

388 Tommy Lee Jones
```

Assume that `str` holds a user set string. Write code to open the file and print out each integer value associated with each person whose name contains the given name. Thus, if `str` held "Jones", 421 and 388 should be printed. If `str` held "Green" nothing should be printed. Although Alan Greenspan contains the string, it does not contain it as a name. Use string functions as you write your code.

10. (10 pts) Given the class

```
class String
{
    private:
        int length;
        char *sptr;
    public:
        String(char *);

};
```

This class is used to maintain a string variable. The **sptr** member will point to the space containing the string and will never waste space.

Overload the assignment operator so that one can perform the following code without using strcpy.

```
String s1('hello');
String s2('goodbye');

s1 = s2;
```

11. (60 pts)

- (a) Write a class to maintain information about a **state**. It should contain data members to hold the population and the name of the state. It should have a constructor that takes parameters for each data member. There should be a function member to print out the information nicely formatted. There should also be a function member to allow the population to be updated.
- (b) Write a class to maintain information about a **city**. It should contain data members to hold the population, the name of the city, and the tax rate of the city. It should have a constructor that takes parameters for each data member. There should be a function member to print out the information nicely formatted. There should be a function member to allow the population to be updated. There should also be a function member to allow the tax rate to be updated. Reuse code wherever possible.
- (c) Write a class to maintain information about a **country**. It should contain data members to keep track of the number of states in the country and an array of those states. Write a constructor to initialize the data members. Have a member function to print out the information nicely formatted. Have a member function to allow the country to create a new state. Have a member function to manipulate the information in one of its states. Reuse code wherever possible.
- (d) Write code to allow one to maintain a heterogeneous list of either **cities** or **states**. The most the list will ever contain is three objects. Set the first object in the list to a **city** (you can make up the data member values). Set the second object in the list to a **state** (you can make up the data member values). Set the third object in the list to a **city** (you can make up the data member values). Show code to print out the information in the list using virtual functions.

