

Name: _____

SSN: _____

Login: _____

CS 12 - Midterm May 22, 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	9	
Problem 2	14	
Problem 3	10	
Problem 4	10	
Problem 5	10	
Problem 6	10	
Problem 7	12	
Problem 8	25	
TOTAL	100	

Name: _____

SSN: _____

Login: _____

1. (9 pts) Define the following terms.

(a) Recursion

(b) Pointer

(c) Private control access of a class

2. (14 pts) True/False (please write out “True” and “False”, do not just put a T or F).

- (a) A string is a character array.
- (b) The %d directive in printf allows you to print out a double value.
- (c) Data members of a class never need to be passed as parameters to function members of that class.
- (d) Structures and classes are by default passed-by-reference.
- (e) Modifying a formal parameter and having it affect the actual parameter occurs when using pass-by-reference.
- (f) An ofstream is read-only.
- (g) The actual name of a file is used when writing code statements to access that file.

3. (10 pts) Show the values of each element in the array *A* after the following code has been performed.

```
int A[5] = {22, 5, 3, 16, 9};
int *ip = A;
*ip = 2;
ip++;
(*ip)++;
ip += 2;
*ip += *(ip - 1);
ip++;
*ip += 4;
```

4. (10 pts) Given the following code:

```
float x, y;  
cout << "Enter two floating-point numbers: ";  
cin >> x >> y;
```

Write code to print each value on a separate line, right justified, so that they show one digit to the right of the decimal point and line up at the decimal.

(a) Use the iomanipulator functions.

(b) Use printf.

5. (10 pts) Given the following code:

```
struct this_and_that {  
    double x;  
    int *ip;  
    char name[20];  
}something;  
  
int scores[10][3];  
this_and_that list[20];
```

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

- | | | | |
|----------|-----------------|---|---------------------------------|
| _____ 1 | something | A | integer |
| _____ 2 | this_and_that | B | double |
| _____ 3 | something.ip | C | character |
| _____ 4 | *(something.ip) | D | character array |
| _____ 5 | list.x | E | integer array |
| _____ 6 | name | F | 2-dimensional array of integers |
| _____ 7 | list[3].name | G | this_and_that |
| _____ 8 | scores[0][0] | H | character pointer |
| _____ 9 | scores[2] | I | integer pointer |
| _____ 10 | list[5].name[2] | J | invalid expression |

6. (10 pts) State what is wrong with the following code segments.

- (a)
- ```
int num;
char *name;

cout << 'Enter a number: ';
cin >> num;

for(int i=0; i < num; i++){
 cout << 'Enter name: ';
 cin.getline(name, 30, '\n');
 cout << 'You entered ' << name << endl;
}
```
- (b)
- ```
// using pointers to pass by reference
void func(double *number)
{
    // change number to user specified value
    cout << 'Enter a floating-point number: ';
    cin >> num;
}

void main()
{
    double x = 7.2;
    cout << x << endl;
    func(num);
    cout << x << endl;
}
```
- (c)
- ```
class X
{
 int a;

 void print();
}
```

```
(d) char filename[20];
 int value;

 cout << "Enter file name: ";
 cin >> filename;

 ifstream infile("filename");

 // echo each number in file to the screen
 while(!infile.eof()){
 cin >> value;
 cout << value << endl;
 }

 infile.close();
```

7. (12 pts) Write a recursive function that takes in one integer and prints out the digits of that integer in reverse. For example, if the number was "12478", the function would print "87421".

8. (25 pts) Write a class and its member function definitions (the code for the functions) to maintain the following. Use string functions wherever possible to receive full credit.

**str** Data member that will hold a user specified string of up to 80 characters maximum.

**strd** Data member that will hold the double of the string. The double of a string is defined to be the string immediately followed by a duplicate copy of the string. For example, if the string was "Hello", the double of the string would be "HelloHello". This string should not ever waste any space.

**Constructor** Initialize members.

**Destructor** Perform clean-up.

**Set** Allow the user to enter a string to be held in *str*, set *strd* to be its string double.

**Print** Print *str* and *strd* nicely.