

Physics 234: Computational Physics

In-class Midterm Exam

Wednesday, February 11, 2009

Student's Name: _____

Fill-in-the-blank and multiple choice questions (20 points)

Mark your answers on the exam sheet in blue or black ink. Please be clear about your selections. Each question is worth one point unless otherwise indicated.

1. Rank these integer types in order of their size (from smallest to largest; write 1, 2, 3, 4 in the spaces provided).

___ char

___ int

___ short int

___ long int

2. Which of the following are legitimate floating-point types? Indicate in the affirmative by circling your choices.

short float, float, unsigned float, long float,
short double, double, unsigned double, long double

3. (3 points) (i) In the code listing below, circle every *scope operator* and every *increment operator*. (ii) Add all the missing semicolons necessary to make the program syntactically correct. (iii) Write the terminal output produced by this program in the box provided.

```

#include <iostream>
using std::cout
using std::endl

int main()
{
    for (int i = 0, j = 1; j != 11; i++, j +=2)
        cout << i << " (" << j << ") "
    cout << endl
    return 0
}

```

terminal output:

4. Write the terminal output from the following code snippet in the box below.

```

int n = 10;
cout << "countdown: ";
while (n > 0) cout << n-- << ", "; do
{ cout << n << endl; } while (n > 0);

```

terminal output:

5. Which of the three integer variables does `::k` refer to? Circle the corresponding variable definition.

```

int k = 0;
int main()
{
    int k = 1;
    {
        int k = 2;
        ::k;
    }
    return 0;
}

```

6. Match each of the following object declarations with the most appropriate description. Write the corresponding letter in the space provided.

(a) `int* p;` (b) `double x;` (c) `char a;` (d) `char r[5];`

- ___ high-precision floating point number
- ___ integer type of exactly one byte in length
- ___ pointer to a fundamental type
- ___ 40 contiguous bits of storage

7. What are the final values of `i` and `x` after the following two statements are executed. (Be sure to express your answers in a notation consistent with the type of each variable.)

```
int i = 0;
const double x = 4*++i + 2 * 3.0+8 / 5+-4 +i++;
```

8. Suppose that your computer represents `signed char` internally using the two's complement scheme. Carry out the addition `a+b` by hand and fill in the empty boxes.

```
signed char a = 5;
signed char b = -27;
signed char c = a+b;
```

	<i>decimal</i>	<i>two's complement</i>																									
Hint:	6	$(00000110)_2$																									
	27	$(00011011)_2$	+																								
	-29	$(11100011)_2$																									
			<table border="1" style="border-collapse: collapse; text-align: center;"> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> <tr><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr> </table>	0	0	0	0	0	1	0	1	1								1							
0	0	0	0	0	1	0	1																				
1																											
1																											

9. (2 points) Given the C array definitions,

```
float p[4] = { 0.125F, 0.25F, 0.5F, 1.0F };
float q[4] = { 0.125F, 0.25F, 0.5F, 1.0F };
```

what do each of the following equalities evaluate to? In each case, circle either true (t), false (f), or unpredictable (u).

```

(t / f / u) p == q;
(t / f / u) *p == *q;
(t / f / u) &p == &q;
(t / f / u) *(p+4) == *(q+4);

```

10. Choose the best description of what the mystery function does.

```

void mystery_func(double out[],
                  const double inL[], const double inR[],
                  int N, int M)
{
    for (int i = 0; i < N; ++i)
        for (int j = 0; j < M; ++j)
            out[i + N*j] = inL[i]*inR[j];
}

int main()
{
    double u[3] = { 1.0, -2.0, 5.0 };
    double v[4] = { -3.0, -4.0, 1.5, 2.25 };

    double A[12];
    mystery_func(A,u,v,3,4);
    return 0;
}

```

- (a) computes the outer product $A_{ij} = u_i v_j$ with the elements of the $N \times M$ matrix A stored in *row-major* order in a single array
 - (b) computes the outer product $A_{ij} = u_i v_j$ with the elements of the $N \times M$ matrix A stored in *column-major* order in a single array
 - (c) computes the inner product $A = \sum_i u_i v_i$
 - (d) program won't compile because of a syntax error
11. Which argument list for `swap` causes this program to output `time-space` rather than `space-time`?

- (a) `void swap(char p[], char q[])`

(b) `void swap(char const p[], char const q[])`

(c) `void swap(char* &p, char* &q)`

```
{
    char* tmp = q;
    q = p;
    p = tmp;
}

char text1[] = "space";
char text2[] = "time";

int main()
{
    char *a = text1;
    char *b = text2;
    swap(a,b);
    cout << a << "-" << b << endl;
    return 0;
}
```

12. The following code compiles but generates a runtime error. Uncomment as few statements as possible so that the the program runs without crashing. (Just scratch out the offending `//` markers.)

```
double *p;
// double x = 1.0;
// int y = 2;
// p = &x;
// p += y;
*p = 2.0;
assert(*p == 2.0);
```

13. Suppose I invoke a program by typing this at the command line:

```
$ ./myprog file1.dat file2.dat
```

Circle the two letters that are accessed as `argv[1][1]` and `argv[2][6]` from within `main`.

14. What is the best mathematical interpretation of the calculation carried out by the function call `special(sin,-0.25*M_PI)`?

```
double special( double (&f) (double), double x )
{
    const double a = f(x);
    const double b = ( a < 0 ? -a : a );
    return b*b*b;
}
```

- (a) $\sin^3(-\pi/4)$
- (b) $3\sin(-\pi/4)$
- (c) $(-\pi/4)^3 \sin(1)$
- (d) $|\sin(-\pi/4)|^3$

15. (3 points) Having defined a C string

```
const char message[] = "The total value of the sum is ";
```

we want to compose a function named `write` such that the function call `write(message)`; produces the same output as the statement `cout << message << endl;`. The function we come up with is the following.

```
void write(const char* const p)
{
    while (*p != '\0') cout << *p++;
    cout << endl;
}
```

Unfortunately, this first attempt isn't quite right.

- (i) Cross out the one `const` modifier that prevents this program from working correctly. (ii) Will `write(message)`; still execute if we remove both `const` modifiers? (yes / no) (iii) Will the program work identically if `while(*p != '\0')` is replaced by `while(*p)`? (yes / no)