University of Waterloo
Department of Systems Design Engineering
SY DE 121 – Digital Computation
Midterm Exam
2:30 – 4:00pm, Wednesday, October 26, 2011

This midterm examination counts for 20% of your total course grade.

NAME (PRINT): Solutions

STUDENT ID NO.: _____

SIGNATURE: _____

EXAMINATION GUIDELINES

a) No candidate will be permitted to leave the examination room during the last 15 minutes of the examination. All candidates remaining during the last 15 minutes of the examination period must remain at their desks until an instructor has collected their papers.

b) No electronic calculators, computers or other computing devices may be used during the test. All you need are erasers, pencils and/or pens.

c) This is a closed-book test. You must not refer to books or notes during the test.

d) Candidates are requested to write only in spaces provided. If you write any rough work on the question paper, cross it out with a large "X". Rough work will not be marked. You may use the back of the pages for rough work.

e) If you pull out individual papers from this exam, make sure that your name and id are written at the top of the loose pages. You are permitted to bring a stapler into the exam room to restaple your exam. Otherwise, ensure that any loose pages are inserted into the exam.

f) A candidate must report to a supervisor before leaving the examination room.

g) Exam papers must be handed in promptly when the signal is given.

Programming Style - Important

When answering questions that ask you to write programs or functions, you should follow the style guidelines that we have discussed in class, unless otherwise directed. Always leave out the file-header comment that gives the file name, your name and ID, program description and the date. You do not need to fully document function prototypes, however, you should use whatever concise documentation you deem necessary to help explain your solution. Do not be concerned about changing output format unless specifically asked i.e., default output format is normally acceptable.

You should be mindful of using appropriate style whenever asked to write your own C++ code.

Some predefined functions in C++:

Prototype	Description	Example	Value of Example
`int abs(int i);`	absolute value	abs(-2)	2
`double fabs(double x);`	absolute value	fabs(-2.4)	2.400000000
`double pow(double x, double y);`	power x^y	pow(2,3) $\Rightarrow 2^3$ pow(2,0.5) $\Rightarrow \sqrt{2}$	8.0000 2.0000
`double sqrt(double x);`	square root	sqrt (3)	1.732050808
`double exp(double x);`	exponential e^x	exp(1)	2.718281828 $e \approx 2.718281828$
`double log(double x);`	base e logarithm	log (2.7183) [$2.7183 \approx e$]	1.000006685 [$1.000006685 \approx 1$]
`double log10(double x);`	base 10 logarithm	log10(100)	2.0000
`double sin(double x);`	sine of radian angle	sin(0.7854) [$0.7854 \approx \pi/4$]	0.707108080 [$0.7071 \approx 1/\sqrt{2}$]
`double cos(double x);`	cosine of radian angle	cos(3.14) [$3.14 \approx \pi$]	-0.999998732 [-0.999998732 ≈ -1]
`int rand();`	random number generator	int a = rand( );	random number between 1 and RAND_MAX

QUESTION 1 [Total: *5 marks*] MULTIPLE CHOICE **For each question, circle the letter corresponding to the most appropriate answer. If you answer with 'None of the above' then indicate the correct answer in the blank line provided.**

1) Augusta Ada is known for:
   a) the concept that a set of instructions could be performed repeatedly
   b) creating the programming language Ada
   c) building the first mechanical calculator
   d) inventing the game of poker
   e) None of the above Correct answer? _____

2) The Apple II had how much RAM?
   a) 16kb
   b) 32Mb
   c) 16Mb
   d) 64kb
   e) None of the above Correct answer? _____

3) How many bits are in a byte?
   a) 16
   b) 32
   c) 64
   d) 128
   e) None of the above Correct answer? _____ 8 bits in a byte _____

4)
```
int a = 40;
double b = 8.9;
int c = 1;
int d = 2;
double q = ( a > b ? ( c > d ? exp(c) : exp(d) ) : 100 );
```

For the above piece of code, `q` is evaluated as:
   a) 40.0
   b) 8.9
   c) 100.0
   d) 1.0
   e) 2.0
   f) Will not compile
   g) None of the above Correct answer? _____ exp(2) _____

5) Which `for` loop will iterate exactly 11 times? (If two or more loops iterate 11 times , circle all correct answers).
```
a) for ( i=100; i<=110; ++i )
b) for ( i=100; i>80; i-=2 )
c) for ( i=54; i == 74; i+=2 )
d) for ( i= -100; i > -67; i+=3 )
e) None of the above.
```

Question	1	2	3	Total
Marks	/5	/20	/20	/ 45

QUESTION 2 (Total: *20 marks*) PREDICT OUTPUT

For each code or code fragment, determine the output. Assume all code is compilable.

(a) [2 marks]

```
int a = 5;
int b = 8;
int c = 7;

if ( a < 4 )
   cout << "Nirvana" << endl;
else if ( a <= 6 && c >= 8 )
     cout << "Pearl Jam" << endl;
else if ( a >= 5 || b < 8 )
     cout << "Red Hot Chili Peppers" << endl;
else if ( a < 10 && b > 0 && c == 7 )
     cout << "Smashing Pumpkins" << endl;
else if ( b == 8 )
     cout << "Soundgarden" << endl;
else
    cout << "Green Day" << endl;
```

Red Hot Chili Peppers

(b) [4 marks]

```
int x = 4, y = 8, z = 7;
if ( x++ > 4 )
   cout << "Prince" << endl;
else if ( x <= 6 && ++z > 8 )
     cout << "U2" << endl;
else if ( x > 5 || y-- < 8 )
{
     cout << "Van Halen" << endl;
     cout << "Def Leppard" << endl;
}
else if ( x < 10 && y > 0 && z-- == 7 )
     cout << "Guns'n'Roses" << endl;
else if ( y == 8 )
     cout << "Billy Joel" << endl;
else
     cout << "The Police" << endl;
cout << "x " << x << " y " << y << " z " << z << endl;
```

The Police
x 5 y 7 z 7

(c) [7 marks]

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

const int FALSE = 0;
const int TRUE = !FALSE;

int rock( int & b );
bool jazz( int & a, int a );
bool blues( int & b, int & d, int c );

int main()
{
    int microphone = 5;
    int bass = 3;
    int guitar = 12;

    if ( blues( microphone, guitar, bass ) )
       cout << "hit music" << endl;
    else
        cout << "bad music" << endl;
    cout << "microphone: " << microphone << endl;
    cout << "guitar: " << guitar << endl;
    cout << "bass: " << bass << endl << endl;

    cout << endl;
    system("PAUSE");
    return 0;
}
int rock( int & a )
{
    a+=3;
    cout << "a is: " << a << endl;
    return a;
}
bool jazz( int & b, int c )
{
       b++;
       c++;
       int val = rock(b) + b + c;
       cout << "b is: " << b << endl;
       cout << "c is: " << c << endl;
       return val;
}
bool blues( int & b, int & c, int d )
{
     bool flag = FALSE;
     if ( jazz(c,b) && b == c && d++ == 1 )
        flag = TRUE;
     else
        flag = FALSE;
     cout << "b " << b << " c " << c << " d " << d << endl;
     return flag;
}
```

```
a is: 16
b is: 16
c is: 6
b 5 c 16 d 3
bad music
microphone: 5
guitar: 16
bass: 3
```

(d) [4 marks]

```
    int m = 0;
    int n = 0;
    for ( m = 10, n = 10; m < 20 && n > 2; m++, n-- )
        cout << " m " << m << " n " << n << endl;
    cout << endl;
```

m 10 n 10
m 11 n 9
m 12 n 8
m 13 n 7
m 14 n 6
m 15 n 5
m 16 n 4
m 17 n 3

(e) [3 marks]

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

void whatishappening( int );

int main()
{
    whatishappening(54321);
    cout << endl;

    system("PAUSE");
    return 1;
}

void whatishappening( int a )
{
     if ( a/10 )
        whatishappening(a/10);
     cout << ( a%10 ) << "*";
}
```

5*4*3*2*1*

QUESTION 3 (Total: *20 marks*) PROGRAM IMPLEMENTATION

Car Race! Write a complete working program that will simulate the race of two cars around their own circular track. READ THROUGH BOTH PAGES FOR THIS PROBLEM FIRST!

(a) [8 marks] Place the main program below. This should be fully working code, so all necessary libraries, function declarations, etc. should be included (documentation not requried). The main program must do the following:
(i) Create double variables for each vehicle's distance (`distA`, `distB`) and the radius of each track (`radA`, `radB`).
(ii) Each track's radius must be set by user input (via `set_rad` function - see (b) below).
(iii) Each car will run their own race around the circular track (via the `race` function - see (c) below).
(iv) Display the speed and total time (`timeA`, `timeB`) for each car.
(v) Determine and indicate the winner of the race.

```
#include <iostream>
#include <ctime>
#include <cstdlib>
#include <cmath>
using namespace std;

const double PI = 4.0*atan(1.0);

void set_rad( double & radius );
double race( const double race_distance );
double circumference( const double radius );

int main()
{
   double distA = 0.0;
   double distB = 0.0;
   double radA = 0.0;
   double radB = 0.0;

   cout << "Set radius for car A" << endl;
   set_rad( radA );
   cout << "Set radius for car B" << endl;
   set_rad( radB );

   distA = circumference( radA );
   distB = circumference( radB );

   srand( time(NULL) );
   double timeA = race( distA );
   double timeB = race( distB );

   cout << "Car A speed " << distA/timeA << " meters per second." << endl;
   cout << "Car B speed " << distB/timeB << " meters per second." << endl;
   cout << "Car A time " << timeA << " seconds." << endl;
   cout << "Car B time " << timeB << " seconds." << endl;

   if ( timeA < timeB )
     cout << "Car A winner" << endl;
   else
     cout << "Car B winner" << endl;

   system("PAUSE");
   return 0;
}
```

(b) [4 marks] Write the function body for `set_rad`. The function must make sure that the radius is greater than 0. Iterate (with the **best** choice of a loop) until the user enters a value greater than zero.

```
// use a 'do-while' loop since (a) you don't know when the loop will end
//   and (b) you want the loop to execute at least once
void set_rad( double & radius )
{
     double set_rad = 0.0;
     do
     {
       cout << "Enter radius greater than zero: ";
       cin >> radius;
     } while ( radius <= 0.0 );
     return;
}
```

(c) [6 marks] Write the `race` function. For each second of time, the car will travel a random distance between 0 and 1. Keep accumulating distance until the car has gone once around the track (note `circumference` function below). Since a car will overshoot the finish line during any second, then determine the fraction of a second it took to reach the finish line to determine the overall time for the car to complete the race. (Note: This last point is a bit tricky, so if you don't understand it, better move on.)

```
double race( const double totaldistance )
{
    double totaltime = 0.0;
    double val = 0.0;
    double dist = 0.0;
    while ( dist < totaldistance )
    {
          // 'val' is a double, so convert 'RAND_MAX' to double
          // 'val' will be a random number between 0 and 1
          val = rand()/double(RAND_MAX);
          dist += val;
          totaltime++;
          //cout << totaltime << " " << dist << " " << val << endl;
    }
    double speed = dist/totaltime;
    //cout << "totaltime: " << totaltime << " dist " << dist
    //     << " totaldistance " << totaldistance << endl;
    totaltime -= (dist - totaldistance)/val;
    //cout << "after " << totaltime << " with val: " << val << endl;

    return totaltime;
}
```

(d) [2 marks] Write the 'circumference' function to calculate the total distance around the track.

```
double circumference( const double radius )
{
       return 2 * PI * radius;
}
```