**Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Score:\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**ECE 2036 Test 1**

Open book and notes, PCs and tablets allowed, but no Internet Access and code cannot be run on a PC.

1. (*5%)* Fill in the typical size (per textbook and slides) in bytes of the following C/C++ variable types:

char \_\_*\_*\_\_ bytes int \_\_\_\_\_\_bytes long long \_\_\_*\_\_* bytes

float array[100] \_\_*\_\_\_\_*\_\_ bytes C style string with 60 characters \_\_\_\_\_\_ bytes

1. (*5%)* Where is the keyword “*default*” used in C/C++ and what does it do?
2. (*5%)* A \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is used by compilers to store subroutine or function return addresses along with any local variables. It supports nested subroutine calls and recursion. The acronym used to describe this type of data structure is \_\_\_\_\_\_\_\_\_
3. (*5%)* As a safe programming practice, when should “*const”* be used and what exactly does it do?
4. (*10%)* Write a C/C++ *void* function called *limit10* with one argument that changes the argument to ten in the calling program whenever the argument is greater than 10. The value of the argument is the address of an integer. Show an example call with the integer variable *j*. Assume *j* is already declared an integer.
5. (*10%)* Write a C/C++ *for* statement using an integer loop control variable “*i*” that would add the first one hundred elements of an existing 1D integer array called “*values”* using an existinginteger variable *“sum”* that is already initialized to zero*. ”i”* must go out of scope when the *for* loop exits.
6. (*20%)* Write the output produced by this program exactly as it will appear on the output device.

#include <iostream>

using namespace std;

class test

{

public:

test(int x);

void p(test y);

int getw()

{

return w;

}

private:

int w;

};

test::test(int x)

: w(x)

{

}

void test::p(test y)

{

w = y.w \* -2 ;

}

int main(int argc, \_TCHAR\* argv[])

{

test A(1);

for (int i=1; i<=9; i++){

A.p(A);

cout << A.getw() << " ";

if (i%4==0) cout<< endl;

}

}

1. (*20%)* Write the output that is produced by this C/C++ program.

#include <iostream>

using namespace std;

int main(int argc, \_TCHAR\* argv[])

{

int a[9]={0,1,2,3,4,5,6,7,8};

int \*aptr;

aptr = &a[1];

a[2] = a[2] + 1;

a[3] = a[2] + aptr[4];

++aptr;

(\*aptr)--;

(\*(++aptr))++;

cout << a[1] << a[2] << a[3] << a[4] << \*(++aptr) << endl;

}

The following questions refer to the mbed C++ program on the next page. A pushbutton is connected from P8 to gnd.

1. (*5%*) What does the *func(a)* function do on the mbed module? Describe in detail what happens when it is called in relation to a’s binary value.
2. (*5%*) What external event must occur before the *while(1)* loop starts?
3. (*10%*) What would you notice when this program is compiled and starts running the *while(1)* loop on the mbed module? Describe the exact initial state, and all of states it will cycle through. If you happen to recall the name of the Si Fi characters that use a similar effect include that and a shorter description of all of the states it cycles through can be used in your answer (but still include the exact initial state).

#include "mbed.h"

DigitalOut l1(LED1);

DigitalOut l2(LED2);

DigitalOut l3(LED3);

DigitalOut l4(LED4);

DigitalIn a(p8);

void func( int a)

{

l1 = a & 0x01;

l2 = (a>>1) & 0x01;

l3 = (a>>2) & 0x01;

l4 = (a>>3) & 0x01;

}

int main()

{

int b = 0;

bool d = 1;

a.mode(PullUp);

wait(.001);

while(a!=0) { }

while(1) {

if (d==0) {

b = b << 1;

if (b==16) {

d=1;

b=8;

}

} else {

b = b >> 1;

if (b==0) {

d=0;

b=1;

}

};

func(b);

wait(0.133);

}

}