Sec: 1 2 (circle) ECE-1021 Spring 2004 Exam #2 18 March 2004

NAME: KEY Score:______/100

Unless specified otherwise, assume that:

- The variables i, j, k, l, m, and n are declared as int
- The variables u, v, w, x, y, and z are declared as doubles.
- Any reference to a "floating point" value is to be taken as a type double.
- Any reference to an "integer" value is to be taken as a type int.

IEEE-574 Single-precision floating point standard: 32 bits total including an 8-bit exponent.
IEEE-574 Double-precision floating point standard: 64 bits total including an 11-bit exponent.

Multiple Choice (2 points each).

1) Which of the following statements is false?

[✓ a) Negative values cannot be interpreted as logical values.]
b) A logical operator will always return a value of 0 or 1.
c) A value is a logical FALSE only if it is equal to 0.
d) A value is a logical TRUE if it is equal to –1.

2) Which of the following code fragments will not returns the same result as r = m % n; ?

```
a) while(m>n) r = (m -= n);
```
[✓ b)] `r = n*(m/n);`
```
c) r = m - n*(m/n);
d) for(r = m; n < r; r -= n);
```

3) What must be true of an else statement?

a) It must contain a logical test.
b) It is associated with the closest if() statement above it.
[✓ c) It must appear immediately after an if() statement in the program structure.]
d) It must be properly indented for the compiler to determine which if() statement is controlling it.

4) What is the value of the following expression?

```
!( (4 - 4%3) < 3 || (6/4 > 1) )
```

$$\underbrace{!\big(\underbrace{\underbrace{(4 - \underbrace{4\%3}_{1})}_{3} < 3}_{F} \;||\; \underbrace{\underbrace{(6/4)}_{1} > 1}_{F}\big)}_{!F \rightsquigarrow T=1}$$

a) 4.
b) 2.
[✓ c) 1.]
d) 0.

5) If j = 0, k = 2 and m = 15, what value is stored in n?

$$n = 8*\underbrace{(\overset{2}{k}\ \&\&\ \overset{15}{m})}_{\underbrace{8*1}_{8}} + 16*\underbrace{\underbrace{(j < \overset{2/15}{k/m})}_{0<0}}_{\underbrace{F}_{0}} + 32*\underbrace{(\underset{\underset{F}{0}}{j}\ ||\ (\underbrace{!m}_{\underset{F}{!15}}))}_{0} + \underbrace{(m/k)}_{\underbrace{15/2}_{7}};$$

a) 8.
[✓ b)] 15.
c) 15.5.
d) 31.5.

6) The "golden rules" regarding parameterized macros serve what primary purpose?

a) Make the macro shorter and easier to debug.
[✓ b)] Force complete evaluation of macro arguments and macro results.
c) Satisfy the syntax requirements of the preprocessor.
d) Permit macros to be "typeless", unlike their function counterparts.

7) What feature must a function have to be classified as a recursive function?

a) A path from the start of the function to the return statement that does not involve a call to itself. SAME NOTE AS (D)
[✓ b)] At least one execution path that involves a call to itself.
c) A function pointer that is initialized by something other than the invoking function.
d) A "base case" return path. STILL CAN BE RECURSIVE, JUST NOT SUCCESSFULL

8) You are given a function, FunctionA(), that always calls FunctionB(). In developing FunctionB() you find it convenient to invoke FunctionA(). What steps must you take to prevent these two functions from calling each other indefinitely?

a) Make sure that FunctionB() is able to call itself.
[✓ b)] Make sure that there is a path through FunctionB() that does not call FunctionA().
c) Make sure that there is a path through FunctionB() that does not call itself.
d) Resist the temptation to call the invoking function – this is guaranteed to result in an infinite loop.

9) Which of the following would be a potential glossary entry for "recursion"?

a) A function or process that does not behave in a predictable fashion.
[✓ b)] See "recursion".
c) A function or process with an indefinite terminating condition.
d) A means of describing a problem in terms of similar problems.

10) Which of the following is not a common drawback of recursively-implemented algorithms?

a) They are slower compared to their iterative counterparts.
[✓ b)] They require more time and effort to craft a solution than their iterative counterparts.
c) They use more memory resources than their iterative counterparts.
d) They are more prone to "locking up" than their iterative counterparts.

11) What is the most likely reason that a "nibble" is the size that it is?

a) Because that is exactly half a byte.
b) Because the first processor has a data bus that was this size.
c) Because anything larger would require the use of a number base that is too difficult for humans to work with.
[✓ d)] Because that is the smallest number of bits that can be used to represent all of the decimal digits.

12) What property does two's compliment exploit?

a) That inverting all of the bits in a value produces a one's compliment value.
[✓ b)] The fact that, in a fixed-width binary value consisting of N bits, that 2^N is indistinguishable from zero.
c) That subtraction is equivalent to adding the negative of a number.
d) That addition and subtraction use the same hardware in a computer.

13) How is zero represented in a IEEE standard floating point representation?

a) Zero cannot be exactly represented because of the implied leading one in the mantissa.
b) As a pattern with a one as the second bit from the left.
c) As a pattern of all zeroes except for a leading one.
[✓ d)] As a pattern of all zeroes.

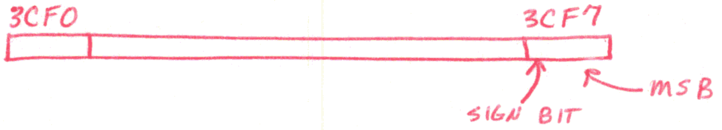
14) Which of the following is not a purpose of a function prototype?

a) To permit the compiler to compile code where it encounters a call to the function prior to encountering the function definition itself.
b) To permit the function definition to be kept in a completely different .c file.
c) To allow the compiler to do type checking on function calls.
[✓ d)] To force the compiler to use the return type specified by the prototype.

15) Which of the following features of the IEEE floating point standard is used to avoid a gap as the representation moves from a normalized to a non-normalized interpretation?

[✓ a)] Two exponent patterns represent the same exponent with the difference being that the mantissa for one is normalized with the mantissa for the other is not.
b) The largest exponent pattern (all 1's) is trapped and treated as a special case.
c) The pattern for an exponent of zero is represented by a leading 0 with the rest of the bits set to 1.
d) The exponent is represented in an offset-binary representation.

16) An eight byte IEEE floating point value has been stored in the range of bytes from address 0x3CF0 to 0x3CF7 according to the Little Endian convention. At what address can the information regarding the sign (positive or negative) be located?



a) 0x3CF0
b) 0x3CF4
[✓ c)] 0x3CF7
d) 0x3CF8

17) The number of distinct ways that a standard deck of fifty-two playing cards can be stacked is 52!, To within 1%, this number is 8E67. Of the following options, which is the smallest width integer (in bytes) that could represent this value?

QUICKEST:

$256^N = 8\times10^{67}$

$N = \dfrac{\ln(8\times10^{67})}{\ln(256)} = 28.1$

a) 16
b) 24
[✓ c)] 32
d) 36

18) What is the value of

```
fives(ceil(sqrt(62.5)))?
```

$\underbrace{\text{ceil}(\underbrace{\text{sqrt(62.5)}}_{7.xxx})}_{8}$

where:

```
int fives(int n) {return( ((n%5) != 0) );}
```

$\underbrace{\underbrace{\underbrace{8\%5}_{}}_{3}\ \ != 0}_{T}$

a) 0
[✓ b)] 1
c) 7
d) 8

19) Which of the following is not an advantage of modularizing a program?

a) The problem can be broken down into self-contained sub-problems. T
[✓ b)] The details of how tasks are performed is readily apparent to all users. F
c) The code can be designed, implemented and tested in manageable pieces. T
d) The code can be reused both in the same project and in future projects. T

20) If you desire to print certain characters using printf(), such as ‘ and “, you must precede them with a special character. What is that character:

a) %
[✓ b)] \
c) /
d) &

ECE-1021 Spring 2004 Exam #2

Short Answer (5 pts each)

21) What value is stored in k?

```
k = ((25/10) / 25)/20;
```

2 / 25

0 / 20

0

22) If a (non-global) variable is not initialized, what value will it have by default?

WHATEVER VALUE WAS ALREADY IN MEMORY AT THAT LOCATION

23) Briefly summarize the key elements of the top-down algorithm development strategy.

DIVIDE AND CONQUER

- BREAK PROBLEM INTO SMALLER (INDEPENDENT, IF POSSIBLE) PROBLEMS
- REPEAT THIS DIVISION PROCESS AS FAR AS NECESSARY UNTIL PROBLEMS ARE SIMPLE ENOUGH TO SOLVE.
- BUILD TOTAL SOLUTION OUT OF SMALLER SOLUTIONS

24) For reliable code, what step should be taken after opening a file and before any other operation is performed on that file? Show a code fragment that performs this step. What are some of the potential ramifications if this is not done?

CHECK TO SEE IF IT OPENED SUCCESSFULLY

```
fp = fopen(          );    // need<stdio.h>
if(NULL == fp)
{
    printf("ERROR MESSAGE");
    exit(1);
}
```

unpredictable performance, including system corruption on some O/S's, CAN result if check not performed.

LONG ANSWER

25) (5 pts) Write a parameterized macro, named **hypotenuse**, that computes the length of the hypotenuse of a right triangle given the lengths of the two sides.

```
#include <math.h> // sqrt()
#define hypotenuse(a,b)  (sqrt((a)*(a)+(b)*(b)) )
```

26) (10 pts) Write a complete function called XtotheY() that takes two positive floating point values and returns the value obtained if the second argument is the exponent to which the first argument is raised. You may NOT use the pow() function. Hint: If you want to get any credit on this problem, do NOT use the pow() function!

$z = x^y$

$\ln(z) = y * \ln(x)$

$z = \exp(y * \ln(x))$

```
#include <math.h> // log(), exp()

double XtotheY(double x, double y)
{
    return (exp (y * log(x)));
}
```

27) (10 pts) Write a complete C function that accepts the following arguments in the following order:

x[ROWS][COLS]

- a pointer to the first element of a two dimensional array of type double. double *x
- an int that is the size of the second dimension of the array int cols
- an int that is the first index for a particular element in the array int r
- an int that is the second index for a particular element in the array int c
- a double that is the value to be stored at the indicated position. double value

For instance, given the following code fragment:

```
double x[10][20];
double y;

y = 3.141592654;
SetElement(x, 20, 5, 15, y);
```

should have the same effect that you would expect from:

```
x[5][15]= y;
```

You may not use any array notation (i.e., you can't use the square brackets) anywhere within your function. You are to use pointers only.

```
void SetElement(double *x, int cols, int r, int c, double value)
{                              // x POINST TO x[0][0]
    x += r*cols;   // x NOW POINTS TO x[r][0]
    x += c;        // x NOW POINTS TO x[r][c]
    *x = value;
    return;
}
```

NOTE: CAN BE DONE ON ONE LINE WITH

```
*(x+r*cols+c) = value;
```

28) (10 pts) Each member of the Fibonacci Sequence is obtained by adding the two prior members. The first two members are defined as themselves. Therefore, the sequence looks like this: {0,1,1,2,3,5,8,13,21,34,....}. Write a complete C function that takes an integer n and that uses recursion to return an int that is the nth member of the Fibonacci Sequence. So fib(0) should return 0 while fib(7) should return 13. What is the largest value of n such that fib(n) can be represented as a 16-bit signed integer.

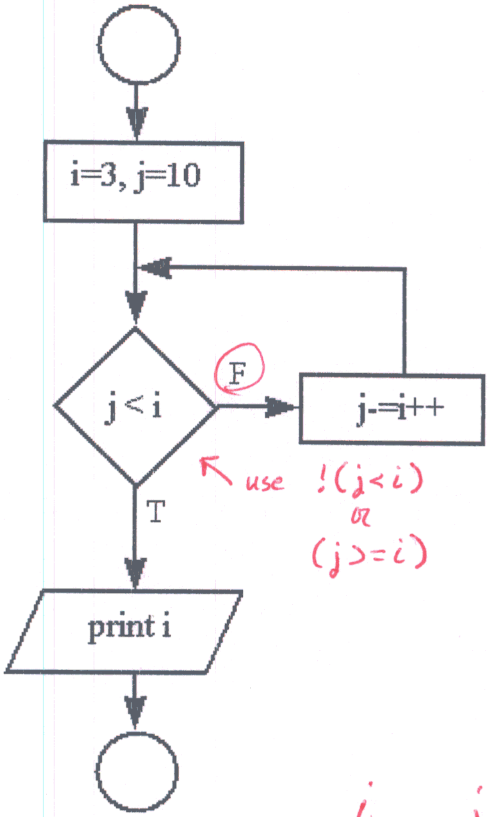
```
int fib(int n)
{
    if (n<2)
        return(n);
    return( fib(n-1) + fib(n-2));
}
```

n	
	21
9	34
	55
	89
	144
	233
	377
	610
	987
	1597
	2584
	4181
	6765
	10946
	17711
23	28657
	46368

LARGEST (POSITIVE) VALUE FOR 16 bit SIGNED INT IS $2^{15} - 1 = \boxed{32767}$

fib(23) IS LARGEST

29) (5 pts) Write a complete C program that implements the following flowchart. What is the value of i and j upon completion?



↖ use !(j<i)
or
(j>=i)

```
#include <stdio.h> // printf()

int main(void)
{
    int i, j;
    i=3;
    j=10;
    while (!(j<i))
        j -= i++;
    printf("%i", i);
    return (0);
}
```

	i	j	j<i
	3	10	F
	4	7	F
final values→	5	3	T

(EXTRA CREDIT)

For large values of n, the factorial of n can be closely approximated by the following equation:

$$n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$$

For instance, for n = 10 the above equation, known as the Stirling Formula, yields a value that is within 1% of the correct answer.

(8 pts) Write a complete C function called fact_digits() that takes a double n as the argument and returns a floating point (double) value that is the number of digits in n!. Your function needs to be able to accept values of n for which n! would be far too large to accommodate even in a variable of long double. For instance, the user might wish to use your function to discover what the first value of n! is that has more than one trillion (10^{12}) digits. A long double, on most compilers, can't even represent a number with 15,000 digits. Hint: How many digits are in 12345? What is the base ten log of 12345?

(2 pts) Within 1%, what IS the first value of n for which n! has more than a trillion digits. Hint: You can do this reasonably quickly on your calculator - but be sure to be finished with the rest of the exam first. Show your work for full credit.

number of digits in x is ceil(log10(x))

$$\log_{10}(n!) = \log_{10}\left(\sqrt{2\pi n}\left(\tfrac{n}{e}\right)^n\right) = \log_{10}(\sqrt{2\pi n}) + \log_{10}\left(\left(\tfrac{n}{e}\right)^n\right)$$

$$= \tfrac{1}{2}\log_{10}(2\pi n) + n * \log_{10}(n/e)$$

$$= (1/2)\log_{10}(2\pi) + (1/2)\log_{10}(n) + n * \log_{10}(1/e) + n * \log_{10}(n)$$

$$- n * \log_{10}(e)$$

$$= (1/2)\log_{10}(2\pi) - n * \log_{10}(e) + (n + 1/2)\log_{10}(n)$$

```
#include <math.h> // log10(), ceil()

double fact_digits(double n)
{
   double digits;
   digits = 0.39909;   // 0.5 * log10(2PI)
   digits += (n+0.5) * log10(n);
   digits -= n*0.43429;   // n*log10(e)
   digits = ceil(digits);   // round up
   return(digits);

}
```

BY FAR dominant term is

digits ≈ $n \cdot \log_{10}(n)$

n	digits ($n \log_{10}(10)$)
10^{12}	$12 \cdot 10^{12}$
10^{11}	$1.1 \cdot 10^{12}$
9×10^{10}	$0.985 \cdot 10^{12}$ (v. close!)
9.5×10^{10}	$1.04 \cdot 10^{12}$
9.25×10^{10}	$1.014 \cdot 10^{12}$

USING FULL EQUATION

9.25×10^{10}	0.97×10^{12}
9.5×10^{10}	1.002×10^{12}

$(9.5 \times 10^{10})!$ HAS 10^{12} DIGITS