Part 1: Multiple Choice (96 points - 3 points per question)

(D) 1. Which statement about C is true?  
(A) Good comments help execution-time performance.  
(B) Text surrounded by /* and */ are processed by preprocessor.  
(C) Lines beginning with a # are processed at execution time.  
(D) none of above

(B) 2. a, b, c, and d are 4 integers and and a = 1, b = 6, c = -8. The value of d after running d = ++a * b % c + b - +c; is  
(A) 16 (B) 17 (C) 18 (D) none of above

(C ) 3. In which header file NULL is defined?  
(A) <io.h>  
(B) <stddef.h>  
(C) <stdlib.h>  
(D) none of above

(A) 4. Which is a multiple selection structure?  
(A) switch (B) if ··· else (C) for (D) while

(B) 5. Which is an example of a ternary operator?  
(A) <= (B) ?: (C) ++ (D) none of above

(B ) 6. If x = 6 and y = -8, what is the value of y for y *= x % y --> -x : y++?  
(A) -40 (B) 64 (C) 72 (D) none of above

(C) 7. How many times will the following program print happy!?  
(i = 1; while ((i *= 2) < 2000) printf("Happy!");  
(A) 8 (B) 9 (C) 10 (D) none of above

(A) 8. Which is the printf conversion specification for long int?  
(A) %ld (B) %lh (C) %lu (D) none of the above

(D) 9. Which function is not defined in <math.h>?  
(A) cos() (B) fabs() (C) exp() (D) ln()

(B) 10. Assume a = 6 and b = 3, which of the following is true?  
(A) a > 3 & b > 12 (B) a * b > 10 & a/b > a % b (C) b > a > 6 (D) none of above

(C) 11. Which is equivalent to if (n != 8)  
(A) if !(n = 8) (B) if !(n - 8) (C) if (n > 8 || n < 8) (D) none of above

(C) 12. Which is illegal type in C?  
(A) int short unsigned (B) long double (C) long unsigned long (D) none of above

(A) 13. Which can be used to convert a lower case letter ch to an upper case?  
(A) ch += 'Z' - 'z' (B) ch += 'a' - 'A' (C) ch -= 'A' - 'a' (D) none of above

(C) 14. Consider the declarations: char c = \'x\'; int i = 6; Give the value and the data type of the expression c * i.  
(A) \"\x14\", char (B) \"\x4E\", char (C) 78, int (D) none of above

(C) 15. int add(int): is an example of a function ___________?  
(A) data type (B) mode type (C) prototype (D) procedure type

(D) 16. Which will not add 1 to a variable?  
(A) a++; (B) a += 1; (C) *p += 1; (D) *p++;

(B) 17. Which can be used to determine the size of an array a[]?  
(A) sizeof(a)/sizeof(a) (B) sizeof(a)/sizeof(a[0]) (C) sizeof(a)/sizeof(a[]) (D) sizeof(a[0])/sizeof(a)

(B) 18. Consider int b[3][2] = {{1}, {2, 3}, {4}}, what is the value of b[1][1]?  
(A) 2 (B) 3 (C) 4 (D) none of above

(C) 19. Given a[3] = [1, 2, 3] and b[3] = [1, 2, 3]. What is the value of a[a[0]] + b[a[0]]?  
(A) 4 (B) 5 (C) 6 (D) 7

(D) 20. Which function is used to seed a new random number sequence?  
(A) rand (B) seed (C) srand (D) srand

(A) 21. int i[] = [10, 20, 30, 40]; int *pa[] = [i+1, i, i+3, i+2]; What is the value of *pa[1]?  
(A) 10 (B) 20 (C) 30 (D) 40

(D) 22. Which is a benefit of functions?  
(A) Good comments help execution-time performance.  
(B) Make a program more efficient.  
(C) Avoid code reuse.  
(D) none of above

(D) 23. How can a random number between -a and a be generated?  
(A) a - rand() % (2 * a + 1); (B) -rand() % (a + 1) + a; (C) a - rand() % (a + 1); (D) none of above

(C) 24. A recursive function is a function that  
(A) returns itself (B) takes a function as an argument (C) calls itself (D) is inside of another function

(D) 25. Unless otherwise specified, an individual array element is passed _______ and an entire array is passed _______.  
(A) call-by-value, call-by-reference (B) call-by-reference, call-by-value (C) call-by-reference, call-by-value (D) call-by-value, call-by-reference

(A) 26. Which is a correct to pass int a[10][10] into a function f?  
(A) f(a) (B) f(a[]) (C) f(a[10]) (D) f(a[10][10])

(B) 27. Which statement is true?  
(A) Arrays cannot contain pointers.  
(B) An integer can be added to a pointer.  
(C) Pointers can simulate call-by-reference.  
(D) none of the above

(D) 28. Which type of variables is not destroyed on exit from the function?  
(A) automatic (B) extern (C) register (D) static

(A) 29. If a = 4.0 and b = 3.0, then what is printed by printf("%d", ceil(sqrt(a + b))));  
(A) 3 (B) 4 (C) 5 (D) none of the above

(D) 30. Which function returns natural logarithm of x?  
(A) exp(x) (B) ln(x) (C) log10(x) (D) none of above

(C) 31. Which value does strcmp("Aloha!", "hello!") possibly return?  
(A) -1 (B) 0 (C) 1 (D) none of above

(C) 32. Which function can be used to get a string from stream?  
(A) fgetc (B) fgets (C) fgets (D) none of above

Part 2: Questions and Answers (78 points)

1. (6 points) Identify and correct the errors in each of the following statements:  

(b) (2 points) int a[8, 8] = 0; i; while(i++ <= 8) b[i, i] = i * i;
(c) (2 points) mul (double x, y) { double x, y; return x * y; }

Ans:

(a) Error: The extra braces are not required for string assignment. The character of a character pointer cannot be changed. The character should be enclosed in single quotation marks.

(b) Error: The array initialization should have braces. b[i][j] should be b[i][j]. The indices of the last element are 7 and 7.
Correction: int b[8, 8] = {0}, i = 0; while(i++ < 8) b[i][j] = i * i;

(c) Error: The function should have a return data type. The parameters should have data types. The parameters should not be declared in the function.
Correction: double mul (double x, double y) {
    return x * y;
}

2. (8 points) Write the result after executing the following program.

```c
message(int i) {
    i % 3 ? printf("Merry! ") : printf("X'mas! ");
}
main() {
    int i = 0;
    while (i++ < 4) printf("Aloha! ");
    printf("\n");
    do printf("Hello! "); while (++i < 7);
    printf("\n");
    for (; i < 12; i++) message(++i);
    printf("\n");
}

Ans:

Aloha! Aloha! Aloha! Aloha!
Hello! Hello!
Merry! Merry! X'mas!
```

3. (a) (5 points) Consider the following recursive function. Rewrite it using iterative (nonrecursive) approach.

```c
int sum(int n) {
    if (n < 1) return 1;
    return sum(n - 1) * (n - 1) + n;
}
```

(b) (5 points) Consider the following function sum. Rewrite it as a recursive function.

```c
int sum (int n) {
    int i, sum = 1;
    for (i = 1; i <= n; i++) sum *= i + 1;
    return sum;
}
```

Ans:

(a) int sum (int n) {
    int i, sum = 1;
    for (i = 1; i <= n; i++) sum = sum * (i - 1) + i;
    return sum;
}

(b) int sum(int n) {
    if (n < 1) return 1;
    return sum(n - 1) * (n + 1);
}

4. (8 points) Write the result after executing the following program.

```c
int func (int a, int b) {
    b *= 2;
    printf("a = %d, b = %d.\n", a, b);
    return --a * (b - 8);
}
```
int sub (int *a, int *b) {
    *a -= 6;
    printf("a = %d, b = %d.\n", *a, *b);
    return *a-- * ++*b;
}
main() {
    int x = 6, y = 8;
    x = func(y, y);
    printf("x = %d, y = %d.\n", x, y);
    x = sub(&y, &y);
    printf("x = %d, y = %d.\n", x, y);
}

Ans:

a = 8, b = 16.
x = 56, y = 8.
a = 2, b = 2.
x = 9, y = 3.

5. (a) (5 points) Write a function factorial (int n) that returns n!.

(b) (5 points) Write the main function that reads x and n and uses the above factorial function and calculate the following series: $x - \frac{x^2}{2!} + \frac{x^3}{3!} - \cdots \frac{x^n}{n!}$.

Ans:

double factorial(int n) {
    int i;
    double f = 1;
    for (i = 1; i <= n; i++) f *= i;
    return f;
}
main() {
    int i, n;
    double x, sum = 0, term;
    printf("Enter n, x: ");
    scanf("%d%lf", &n, &x);
    for (i = 1, term = x; i <= n; i++, term *= -x) sum += term/factorial(i);
    printf("sum = %f\n", sum);
}

6. (a) (6 points) Write a function that passes a string and reverse it.

(b) (8 points) Write a function that passes a string and randomly permutate it.

Ans:

(a) #define SIZE 81
void reverse(char s[]) {
    int len = strlen(s), i;
    char r[SIZE];
    for (i = 0; i < len; i++) r[i] = s[len - i - 1];
    for (i = 0; i < len; i++) s[i] = r[i];
}

(b) void scramble(char s[]) {
    int len = strlen(s), i, j;
    char temp;
    srand(getpid());
    for (i = 0; i < len - 1; i++) {
        j = rand() % len;
        temp = s[j];
        s[j] = s[i];
        s[i] = temp;
    }
}
typedef struct {
    double x, y;
} Point;

(a) (5 points) A function that passes a point and returns an integer (1, 2, 3, or 4) to indicate in which quadrant the point is located.

(b) (4 points) A function that passes two points and returns the distance of them.

(c) (6 points) A function that passes two points and generate the line equation that passes these two points.

(d) (7 points) A function that passes three points and return the area formed by these three points.

Ans:

(a) int quadrant(Point p) {
    if (p.x > 0 && p.y > 0) return 1;
    if (p.x < 0 && p.y > 0) return 2;
    if (p.x < 0 && p.y < 0) return 3;
    if (p.x < 0 && p.y < 0) return 4;
    return 0;
}

(b) double distance(Point p1, Point p2) {
    return sqrt((p1.x - p2.x) * (p1.x - p2.x) + (p1.y - p2.y) * (p1.y - p2.y));
}

(c) void line(Point p1, Point p2) {
    double a = p2.y - p1.y;
    double b = p1.x - p2.x;
    double c = p1.x * (p2.y - p1.y) + p1.y * (p1.x - p2.x);
    printf("%fx + %fy = %f\n", a, b, c);
}

(d) double area(Point p1, Point p2, Point p3) {
    double a = distance(p1, p2), b = distance(p2, p3), c = distance(p3, p1);
    double s = (a + b + c)/2;
    return sqrt(s * (s - a) * (s - b) * (s - c));
}