All multiple choice questions are equally weighted. You can generally assume that code shown in the questions is intended to be syntactically correct, unless something in the question or one of the answers suggests otherwise.

1. **What are the first three lines printed by the following code?**

```
int row = 1, i;
while (row < 10) {
   switch (row % 3) {
   case 0:
       for (i = 1; i <=4; i++)
           printf ("+-");
       break;
   case 1:
       for (i = 1; i <=4; i++)
           printf ("--");
       break;
   case 2:
       for (i = 1; i <=4; i++)
           printf ("++");
       break;
   default:
        for (i = 1; i <=4; i++)
            printf ("-+");
   }
   printf ("\n");
   row = row+1;
}
```

A.   --------
        ++++++++
        -+-+-+-+
B.   +-+-+-+-

        ++++++++
C.   --------
        ++++++++
        +-+-+-+-
D.   -+-+-+-+
        -+-+-+-+
        -+-+-+-+
E.   +-+-+-+-
        +-+-+-+-
        ----

2. **Suppose you have a program that keeps track of deposits, withdrawals, and interest to a bank account. Inside the main() function, there is a variable that keeps track of the current balance in the bank account, declared as follows:**

  double balance;

Which of the following function prototypes could NOT be a function that could be called in main() to figure out the new balance and update the balance variable?:

A. double update_balance(double old_balance, double interest, double amount);
B. int withdraw_money(double *new_balance, double old_balance, double amount);
C. void add_interest(double old_balance, double *new_balance, int days_of_interest, double interest_percent)
D. void withdrawal(double old_balance, double amount, double new_balance, int *is_overdrawn);
E. double calc_moneyleft(double withdraw_amount, double old_balance);

3. **Just after line S of the following program executes, what is the value of stop?**

    #define START 10

    int main (void) {
       double stop, start;
       start = START;
       stop = stop + start; /*line S*/
       return (0);
    }

    A. 0.0
    B. 10
    C. 10.0
    D. unpredictable (although the program will compile without error).
    E. the program will have a syntax error, because start is defined twice (once spelled "START" and once as "start").

4. **If the program fragment below executes, how many times does the word "Hello" get printed?**

    int i, j, k;

    for (i = 0; i < 100; i = i + 10)
      for (k = 0; k < 5; k ++)
        for (j = 1; j <= 1; j++)
        {
          printf("Hello\n");
          i = i-1;
        }

    A. 10
    B. 20
    C. 50
    D. 100
    E. Infinite

5. **What is the value of the expression:**

    x % y + x / y + y % x

    given the following declarations:

    int x = 7;
    int y = 3;

    A. 3
    B. 4
    C. 5
    D. 6
    E. 7

6. **Choose the conditional expression which is true only when the value of x (an int variable) is even and in the range 0 to 6.**

    A. x == 0 && x == 2 && x == 4 && x == 6
    B. x > 0 && x != 1 || x != 3 || x != 5 && x < 7
    C. (x % 2) == 1 && x >= 0 && x <= 6
    D. (x % 2) == 0 && x >= 0 && x <= 6
    E. x == (2 * x) && x >= 0 && x <= 6

7. **#define MAXGRADES 20**
    ...
    int winter[MAXGRADES];
    int spring[MAXGRADES];
    int classSize, i;

    The array winter has been initialized. The variable classSize contains the number of elements of winter which are used (all the used elements are together without gaps starting in the 0th position of the array).

    Which of the following code fragments will correctly copy the used elements of winter to the array spring?

    A.
    spring = winter;

    B.
    for (i = 0; i < classSize && i < MAXGRADES; i ++)
    **   spring[i] = winter[i];**

    C.
    spring[MAXGRADES] = winter[MAXGRADES];

    D.
    spring[classSize] = winter[classSize];

    A. A only
    B. B only
    C. C only
    D. D only
    E. more than one of A, B, C, or D

8. **Among the choices given, the best definition of an "expression" is...**

    A. a series of steps which compute an answer
    B. a mathematical formula
    C. something in a program which has a value
    D. an arithmetic operation
    E. a single statement

9.   **What is the primary job of the Linker?**

    A.   To combine functions, libraries, etc. into an executable program
    B.   To translate source code into object code
    C.   To find syntax errors in a program
    D.   To find semantic errors in a program
    E.   To convert input characters into the proper data types for the variables used for input

10.   **Suppose the function Snork has the following prototype:**
    int Snork(char *p, int x, int *y);

    Which of the following is a legal use of Snork assuming the declarations:
    int i, j;
    char ch;

    A.   i = Snork('A', 10, j);
    B.   j = *Snork(&ch, &i, &j);
    C.   i = Snork(&ch, i, &17);
    D.   Snork(ch, i, &j);
    E.   None of the above

11.   **The setToBiggest function sets both of its input variables (passed as output parameters) to the value of the larger input. Which of the following is the correct function body for**

    void setToBiggest(int *x, int *y){
      BODY }

    A.   if (x > y)
        y = x;
      else
        x = y;
    B.   *y = x;
      *x = y;
    C.   int temp;
      temp = *y;
      if (*x > *y)
        temp = *x;
      *y = temp;
      *x = temp;
    D.   if (*x > *y)
        *x = *y;
      if (*y > *x)
        *y = *x;
    E.   *x = *y;
      if (y < x)
        *y = *x;

12. **typedef struct {**
    char name [MAX_NAME+1];
    int id;
    double score;
    } Student Record;

    StudentRecord a[MAX_STUDENTS];

    What are the types of the following three expressions:
    a[0], &a[5], a[0].name[0]

    A.   StudentRecord, Illegal, char array
    B.   struct, pointer, name
    C.   StudentRecord array, StudentRecord, char
    D.   StudentRecord array, StudentRecord, Illegal
    E.   StudentRecord, StudentRecord*, char

13. **typedef struct {**
    int x, y;
    } Point;

    typedef struct {
    int color;
    Point pos;
    } ColoredPoint;

    MovePoint(ColoredPoint *cp){
    ...
    }

    In the function MovePoint, what is the correct code to move cp to the origin of the coordinate
    system?

    A.   cp->pos.x = 0;
         cp->pos.y = 0;
    B.   cp.pos->x = 0;
         cp.pos->y = 0;
    C.   cp->pos->x = 0;
         cp->pos->y = 0;
    D.   cp.pos.x = 0;
         cp.pos.y = 0;
    E.   cp->pos = {0, 0};

14.
```
void SetToQuestion(char str[]){

}
```

The function SetToQuestion is supposed to set the string str to a string of all '?' characters. The length of the string is not changed.

For example, if we have:
char bob[5] = "ABCD";
then the call SetToQuestion(bob) would make bob the string "????"

Which code correctly implements the body of SetToQuestion?

A.
```
int i=0;
while (str[i] != '\0'){
   i = i + 1;
   str[i] = '?';
}
```
B.
```
int i=0;
while (str[i] != '\0'){
   str[i] = '?';
   i = i + 1;
}
```
C.
```
int i=0;
while (str[i] != '?'){
   str[i] = '?';
   i = i + 1;
}
```
D.
```
int i;

for (i = 0; str[i] != '?'; i++)
   str[i] = '?';
```
E.
```
int i;

for (i = 0; str[i] == '?'; i++)
   str[i] = '\0';
```

15.
```
char str[6];
strcpy(str, "BLACK");
strcpy(str, "RED");
```

What are the values of str[3], str[4], str[5] after the two calls to strcpy?

A.   '\0', '\0', '\0'
B.   '\0', 'K', '\0'
C.   'C', 'K', '\0'
D.   'C', 'K', 'R'
E.   'D', '\0', 'K'

16. **About how many array elements need to be examined when binary search is done on an array of 65536?**

    A. 8
    B. 16
    C. 32
    D. 256
    E. 65536

17. **Study this code fragment, and then answer the question that follows it.**

    ```
    int find_something(int b[], int low, int high){
      int i, pos;

       pos = high;
       i = high;
       while (i >= low){
         if (b[i] > b[pos])
            pos = i;
            i = i - 1;
          }
       return pos;
    }
    ...
    int c[5] = {8, 3, 11, 7, 12};
    find_something(c, 0, 3);
    ....
    ```

    what value is returned from the call to find_something shown just above?

    A. 0
    B. 1
    C. 2
    D. 3
    E. 4

18. **void convert (int total_cents, int *dollars, int *cents){**

    ..
    }

    The function above in meant to convert from a total number of cents into the equivalent dollars and cents. For example, on the call
    convert(337, &d, &c)
    d should be assigned 3 and c assigned 37.

    Which is the correct function body?

    A. cents = total_cents % 100;
        dollars = (total_cents - cents)/100;
    B. dollars = total_cents/100;
        cents = total_cents - 100*dollars;
    C. dollars = total_cents/100;
        cents = total_cents%100;
    D. *dollars = total_cents/100;
        *cents = total_cents%100;
    E. *dollars = *total_cents/100;
        *cents = *total_cents%100;

19. **int x = 4, y = 3;**
    double a, b;
    a = (double) (x / y);
    b = (int) (x / y * x);

    What are the values of a and b after the code fragment executes? [Double values are shown to 2 decimal places.]

    A. 1.33 5.33
    B. 1.33 3
    C. 1.00   5.33
    D. 1.00   4.00
    E. 1.33 4

20. **You have written a program to help you with your birthday shopping.**
    The following structure is implemented. What is printed by executing the code below?

```
#define TRUE 1
#define FALSE 0

typedef struct{
        int age;
        int month_birthday;
        int day_birthday;
        int present_purchased;
        char first_name[20];
} person;

void purchased_present (person *p){
   p->present_purchased = TRUE;
}

person had_birthday (person p){
   p.age = p.age + 1;
   return p;
}

..........

person Sara = {10, 4, 30, FALSE, "Sara"};
person Mike = {40, 1, 10, TRUE, "Mike"};
person NewMike;

purchased_present(&Sara);
NewMike = had_birthday(Mike);
if(Sara.present_purchased == FALSE)
   printf("You need to buy a present for %s\n", Sara.first_name);

printf("%s is %d years old\n", Mike.first_name, Mike.age);
```

    A.   You need to buy a present for Sara.
         Mike is 40 years old.
    B.   You need to buy a present for Sara.
         Mike is 41 years old.
    C.   Mike is 40 years old.
    D.   Mike is 41 years old.
    E.   You need to buy a present for Sara.
         NewMike is 41 years old.

21. **Suppose you want to implement an address book containing information about your friends that**
    you have met in cse142. The information you want to store includes their names, their phone
    numbers, their e-mail addresses, and their street addresses. What would be the best data
    structure for the address book?

    A.   An array of sorted integers
    B.   An array of structures where each structure represents one friend
    C.   A single array containing all the pieces of information for your friends.
    D.   A single structure containing all the pieces of information for your friends.
    E.   Parallel arrays of integers

22. **Study the function rec, then choose the best statement about it.**

```
int rec(int x){
  if (x >= 0)
     return x;
  else
     return rec(-x);
}
```

The function rec...

A. is syntactically incorrect since it references itself.
B. runs forever for some parameter values.
C. computes the absolute value of x.
D. always returns 0.
E. counts the number of binary digits in the number.

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

for (i = 0; i < 6; i = i + 2)
   a[i+2] = a[i];

What is the contents of a after this program fragment executes?

A. 0 1 2 3 4 5 6 7 8
B. 0 1 0 1 2 3 4 5 6
C. 2 3 4 5 6 7 8 7 8
D. 2 3 4 5 6 7 8 9 10
E. 0 1 0 3 0 5 0 7 8

24. **while ((color!=PURPLE) && (count < MAX))...**

Which of the conditions listed is equivalent to the one used in the statement above?

A. ! (color != PURPLE) && ! (count < MAX)
B. ! (color != PURPLE) || ! (count < MAX)
C. ! (color == PURPLE) && ! (count >= MAX)
D. ! ((color == PURPLE) || (count >= MAX))
E. (color ==PURPLE) || (count >= MAX)

25. **A 2-D array holds information about students and their quiz scores. Each row contains information about one student, and each column contains the scores for one quiz. Exception: row 0 contains the perfect possible score for each quiz.**

   The function extraCredit should give one bonus point to every student who did does not already have a perfect score on a particular quiz.

   ```
   void extraCredit (scores[MAXS][QMAX], int quiznum) {
   int s, q;
   int perfect;
   int temp;
   ...
   }
   ```

   Which choice correctly completes the function body for extraCredit?

   A.
   ```
   perfect = scores[0][quiznum];
   for (s = 0; s < MAXS; s++) {
     if (scores[s][quiznum] < perfect)
        scores[s][quiznum]++;
   }
   ```

   B.
   ```
   for (s = 0; s < MAXS; s++) {
     if (scores[quiznum][s] < perfect[quiznum])
        scores[quiznum][s]++;
   }
   ```

   C.
   ```
   s = 0;
   while (s < MAXS) {
     temp = scores[s][quiznum];
     if (temp < scores[s].perfect) {
        temp = temp+1;
     }
     s = s+1;
   }
   ```

   A. A only
   B. B only
   C. C only
   D. A and C only
   E. B and C only

26. **Suppose you see this in a correctly working C program:**

s = fscanf (infile, "%c", &nextChar);

What must be true?

A. infile is the name of a file
B. infile is the name of a variable of type FILE*
C. after the fscanf executes, s contains the next character read from the file (unless there is no data left on the file)

A. A only
B. B only
C. C only
D. All of A, B and C
E. None of A, B, or C

27. **[worth 2.5 M.C. questions]**
    On the preceding pages, you have the listing of a C program (adapted from our Hanly and Koffman textbook, chapter 7). Draw a call graph for this program. Please your answer on this page.

    [For reference, the procedure for making a call graph is given here. It is the same as the one we have been using all quarter.

    For each function in the program, draw a box and put the function name inside the box (don't include any other information, please: no parameters, no comments, nothing but the name). If function A calls function B, put the box for A higher on the page, and draw a line down from the A box to the B box. Start with main at the top.

    If a function is called more than once, draw only one box for it. But draw a line to it from each function that calls it.

    DO Include any functions which are called but which are not defined in the program (such as library functions). If something is NOT a function, it should not have a box; it should not be mentioned at all.]

28. **[worth 4 M.C. questions]**

A game program displays "Objects" of different colors and screen positions.

Write a function called closestRedToOrigin. This function, given an array of Objects and the size of the array, returns the RED Object which is closest to the origin. If there is no RED Object in the array, then return a RED Object at the origin. The contents of the array is unchanged in any case. Structs for use in the problem are shown below.

Note that you are not asked to write a main or a complete program. You may not call ANY functions in your solution, except the pointDistance function shown below, which you must use.

Follow style principles like those expected this quarter for the homework assignments.

```
typedef struct {
  int x, y;
} Point;

typedef struct {
  int color;
  Point pos;
} Object;

double pointDistance (Point p1, Point p2);
```

29. **[worth 3 M.C. questions] This problem is a continuation of the preceding one.**

A game program has produced a file of Objects. Write the main function for a program which reads this file, and displays the RED object closest to the origin. If there was any error reading the file, display no object.

Use closestRedToOrigin (from the preceding problem), to find the RED object closest to the origin (whatever answer it returns is what you should display). In addtion, use the following two functions. Both of these are assumed to be already written: you should NOT write code for them:

/*readObjects
Prompts the user for the name of a previously-created file of information about Objects, opens and reads that file, and places the data in an array of Objects. The second parameter tells the (maximum) size of the array (if there are more objects than that in the file, the extra ones are ignored). As a return value, the function tells how many Objects were placed in the array. If there was any file error, the return value is negative.
*/
int readObjects (Object [], int maxSize);

/*displayObject
Displays one object on the screen
*/
void displayObject (Object Obj);

Aside from the three functions mentioned, you may not call ANY other functions (no, not even printf!)

Follow style principles like those expected this quarter for the homework assignments.