CS 141 – Programming Design II
Midterm Exam #1 - Prof. Reed, Spring 2019

This test is worth 10% of your final grade. This test is open notes (anything on paper), but no electronic resources may be used. When only a program segment is given, you can assume it is otherwise placed in the context of a program that is otherwise correct and includes all declarations and system libraries needed to make it work. This is version one of the test.

For the multiple-choice questions with a bubble next to the answers, you must fill in only one bubble next to the best answer. Some questions have an indicated rectangle where you must write down your answer. Any writing outside of these indicated areas will not count towards your grade. For some problems you will need to know that ASCII ' ' is 32, '0' is 48, 'A' is 65 and 'a' is 97.

There are 14 multiple-choice problems worth 2 points each (14 x 2pts. = 28pts.), and 12 free-response problems worth 6 points each (12 x 6pts. = 72pts.), for a total of 28 + 72 = 100 points.

Multiple-choice problems are 3 points each:

1. What is the connection between the video shown in class of a man eating a bicycle and programming?
   ○ The first step in problem solving is to understand the problem
   ○ We can use functions in a program to solve portions separately
   ○ Computer programming ideas can be used to solve every-day problems
   ○ When stuck on a problem, think of related problems that you've already solved

2. In what sort of situation might multiple if-else statements be preferred over a single switch-case statement?
   ○ When selecting among menu options
   ○ When translating row and column inputs into a board index position
   ○ When assigning a percentage based on the number correct out of 10 problems
   ○ When using a user input letter to determine which of nine tic-tac-toe board pieces to modify

3. Which of the following statements is most accurate regarding all binary numbers that have a right-most digit of 1?
   ○ They are even
   ○ They are odd
   ○ They are all positive numbers
   ○ There is no generalization to be made about them

4. Which of the following statements is most accurate regarding all binary numbers that have a left-most digit of 1?
   ○ They are even
   ○ They are odd
   ○ They are all positive numbers
   ○ There is no generalization to be made about them
5. Consider the number-boxes problem that we discussed in class, used to "guess" some number identified by someone ahead of time:

What determines which number goes in which box as shown at right?
- A number's mod value of some power of 2
- A number's closest multiple of some power of 2
- A number's closest set of prime factors
- A number's value of 1 in some binary number bit-position

6. What is the output of a program that prints the largest positive integer value plus one?
- It causes a compiler error
- It displays a negative number
- It displays a run-time error message
- It displays the largest positive integer value

7. What is the effect of the `exit` statement in a loop?
- It stops running the program
- It reruns the current loop iteration
- It bypasses the rest of the code in the loop and does the next loop iteration
- It bypasses the rest of the code in the loop and resumes execution after the loop

8. What is the effect of the `break` statement in a loop?
- It stops running the program
- It reruns the current loop iteration
- It bypasses the rest of the code in the loop and does the next loop iteration
- It bypasses the rest of the code in the loop and resumes execution after the loop

9. What is the effect of the `continue` statement in a loop?
- It stops running the program
- It reruns the current loop iteration
- It bypasses the rest of the code in the loop and does the next loop iteration
- It bypasses the rest of the code in the loop and resumes execution after the loop

10. Consider code that processes a set of integer test scores in the range 0..100 that are typed in by a user. How many of the following situations would be best implemented using an array?

   I. Display the top five high scores
   II. Find the average of all the scores
   III. Display the scores in ascending order
   IV. Count how many there are of each score

- 1
- 2
- 3
11. Consider the function definition shown below:

```c
int sum( int a, int b)
{
    return a+b;
}
```

What would be the result of attempting to compile and run this code when calling it with:

```c
cout << sum('A',1.5);
```

- It would give a compiler error
- It would compile but would give a run-time error
- It would run and give output of an `integer` value
- It would run and give output of a `floating-point` value

12. Consider the two function definitions shown below:

```c
char sum2( int a, int b)
{
    return a + b;
}

int sum2( int a, int b)
{
    return a + b;
}
```

What would be the result of attempting to compile and run this code when calling it with:

```c
cout << sum2(65,1);
```

- It would give a compiler error
- It would compile but would give a run-time error
- It would run and give output of a `char` value
- It would run and give output of an `integer` value

13. Consider the code shown at left below that attempts to display the multiplication table shown at right below:

```c
for( int row = 0; row < 11; row++) {
    for( col = 0; col < 11; col++) {
        cout << row * col;
    }
}
```

How many of the above lines of code need to be changed to get the table output displayed and formatted as shown? (Multiple code changes on one line count as one.)

- 1
- 2
Consider the code segment shown below. If after the function call to `modify(...)` the value of `counter` has changed, what is the most likely cause?

```cpp
int counter = 0;
char letters[] = "ABCD";
cout << counter;
modify(letters);
cout << counter;
```

- Function `modify` overwrites the end of array `letters`
- `counter` is declared as a global instead of a local variable
- Variable `counter` is shared between two different programs
- Function `modify` itself calls another function, which is the one that changes `counter`

15. What is the output of the following section of code:
   ```cpp
double x = 10;
cout << 3/4 * x;
```
   Write down your answer in the box below. Print legibly.

16. What is the output of the following section of code:
   ```cpp
   int x = 30;
x = x * 2 + 5;
cout << (char)x;
   ```
   Write down your answer in the box below. Print legibly.

17. What is the equivalent decimal value for the binary number `10011001`?
   Write down your answer in the box below. Print legibly.

18. What is the equivalent binary value for the decimal number `61`?
   Write down your answer in the box below. Print legibly.
19. What is output from running the program segment shown below at left? Write down your answer in the box shown at right below. Print legibly.

```cpp
int sum = 0;
for(int row = 0; row < 4; row++) {  // Which comes first?
    for(int col = 0; col < 3; col++) {
        sum = sum + (row * col);
    }
}
cout << sum;
```

Write answer in this box:

20. Indentation is meant to clarify what a program does, but must be done correctly. What is output from running the program segment shown below at left? Write down your answer in the box shown at right below. Print legibly.

```cpp
int num = 4;
if( num >= 3) {
    if( num == 3) {
        cout << "= 3 ";
    } else {
        cout << "< 3 ";
    } cout << "Done";
}
```

Write answer in this box:

21. What is output from running the program segment shown below at left? Write down your answer in the box shown at right below. Print legibly.

```cpp
// Which comes first?
string theChicken = "Chicken";
string theEgg = "egg";
if( theChicken < theEgg) {
    cout << "chicken";
} else {
    cout << "egg";
}
```

Write answer in this box:
22. What is output from running the program segment shown below at left?
Write down your answer in the box shown at right below. Print legibly.

```cpp
int score = 87;
if (90 <= score <= 100) {
    cout << "A";
} else {
    cout << "B";
}
```

Write answer in this box:

```

```

23. Consider the code shown at left below. What is the output of the following C++ program
segment, called with `confuseDriver()`?
Write down your answer in the box at right below. Print legibly.

```cpp
int s=1, y=3;
void confuse1(int a, int b) {
    a++;
    b++;
}
void confuse2(int b, int &s) {
    y = s + 2;
    s = b + 1;
}
void confuse3(int &y, int &s) {
    y = s + 3;
    s++;
}
void confuseDriver() {
    int s=1;
    confuse1( s, y);
    confuse2( s, y);
    confuse3( s, y);
    cout << s+y;
}
```

Write answer in this box:

```

```
24. Consider the code segment shown at left below. What is the output when function `scope()` is called? Write down your answer in the box at right below. Print legibly.

```c++
int x = 4; // global variable
void s1( int y)
{
   cout << x+y;
}
void s2( int y)
{
   x = y++;
   s1( ++x);
}
void scope()
{
   int x = 2;
s2( x);
}
```

Write answer in this box:

25. What is the output from calling function `useValues()` shown at left below? Write down your answer in the box at right below. Print legibly.

```c++
void swap( char &x, char &y)
{
   char temp = x;
   x = y;
   y = temp;
}
void useValues()
{
   const int Max = 6;
   char values[Max] = "ABCDE";
   for (int i=0; i<Max/2; ++i) {
      swap( values[i], values[Max-i-2]);
   }
   cout << values[0] << values[Max-2];
}
```
What is the output from calling function `changeValues()` shown below? Write down your answer in the box at right below. Print legibly.

```cpp
const int Max = 6;

void swap( int &x, int &y)
{
    int temp = x;
    x = y;
    y = temp;
}

void changeValues()
{
    int values[Max] = {6,5,4,1,3,2};
    int x;

    for( int i=0; i < Max-1; i++) {
        x = i+1;
        for( int j=x; j < Max; j++) {
            if( values[j] < values[x]) {
                x = j;
            }
        }
        swap( values[i], values[x]);
    }

    cout << values[ 0] << values[ Max-1];
}
```

Write answer in this box: