CSE 2451 Midterm

Name:

Introduction

duration: 10:20 am-12 pm (100 minutes)
you are allowed to use 1 A4-page cheat-sheet

True/False (15 pts)

True or False for the following statements:

1. Building a program involves processing the source file(s) with the various stages in the exact following order: 1) preprocessing 2) assembling 3) compiling and 4) linking:
2. Operators of the same precedence are guaranteed to have the same associativity:
3. A do while loop will run for at least 1 iteration no matter what value its conditional expression evaluates to:
4. "goto" is a valid variable name:
5. Given an array " int arr[3]; " and a pointer " int *ptr; ", this "ptr = arr;" assignment is valid:

Multiple Choice (60 pts)

The following questions only have one correct choice.

6. Which of the following compiling commands will instruct gcc to apply stricter checking for potential errors and print out warning information?
A. gcc -Wall src.c -o myprog
B. gcc -g src.c -o myprog
C. gcc src.c -o myprog
D. None of the above

7. What is the command used to monitor the **change** of a variable's value when debugging the program with GNU debugger (gdb)?
A. print
B. watch
C. display
D. None of the above

8. What is the value of this expression: (int) ((double) 10/9 -1)
A. -1
B. 0
C. 1
D. 0.111111

9. Given an integer variable x with its value assigned to 3, what's the value of x after the following assignment "x += 0.9;"? (trailing zeros for floating-point values are ignored)
A. 3.9
B. 3.0
C. 3
D. 4

10. Given two integer variables x and y with their values assigned to 3 and 4, respectively. What's the value of x and y after the following assignment "x += (y %= 3);" ?
A. x = 4, y = 1
B. x = 3, y = 1
C. x = 3, y = 0
D. None of the above is correct

11. Given two integer variables x and y with their values assigned to 3 and 4, respectively. What's the value of x and y after the following assignment "x += (y /= 3);" ?
A. x = 4, y = 1
B. x = 3, y = 1
C. x = 3, y = 0
D. None of the above is correct

12. Given two integer variables x and y with their values assigned to 3 and 4, respectively. What's the value of x and y after the following if statement is executed? (+ has higher precedence than >=)

```
1  if ( (x+1>=4) || (++y) ) {
2      y = y − 4;
3  }
```

A. x = 4, y = 1
B. x = 3, y = 1
C. x = 3, y = 0
D. None of the above is correct

13. Which of the following statements about scope is **NOT** true?
A. File scope is larger than function scope.
B. Formal parameters in the function definition has function scope.
C. The label of a goto statement has block scope.
D. Entities belonging to different name spaces would not have name collision even if they have the same identifier and they are in the same scope.

14. Which of the following statements about function in C is **NOT** true?
A. values of local variables declared inside a function is stored in the corresponding stack frame when the function is called.
B. the function call stack is a data structure with First In First Out (FIFO) processing order.
C. the function call stack will create a stack frame for each function call.
D. A function can call itself inside its function definition.

15. What's the value returned by applying the dereference operator on the bar pointer after the following code gets executed?

```
1  int x = 5;
2  int *foo, *bar;
3  foo = bar = &x;
4  *foo += 3;
```

A. 8
B. address of x
C. address of x increased by 3 * sizeof(int)
D. 5

16. Which of the following statements is **NOT** true?
A. typedef enable programmers to declare an alias to existing types in C.
B. An alias declared by typedef follows standard scoping rules, and it belongs to ordinary name space.
C. "typedef char * char_ptr;" declared char_ptr to be a type alias for "a pointer to char" type.
D. "typedef uint unsigned int;" declared uint to be a type alias for "unsigned int" type.

17. Which of the following declaration does not declare a function pointer?
A. char (*ptr)(double, double);
B. char (*ptr)();
C. char *ptr(double, double);
D. char (*ptr)(void);

18. Which of the following declaration declares a pointer to a whole 1-d array of size 5? (a pointer to a whole 1-d array is not a pointer to its 1st element)
A. float (*ptr)[5];
B. float *ptr;
C. float *ptr[5];
D. float (*ptr)();

19. Which of the following statements about function pointer is **NOT** true?
A. A function pointer is a pointer to a function.
B. Functions in C can not return a value of function pointer type.
C. A function call to a specific function can be made by using a function pointer to the function as if it is an alias of the function.
D. A function call to a specific function can be made by using a dereference operator on the associated function pointer to the function.

20. Given the following declaration of an array, and sizeof(double) = 8. How many bytes does this array occupy in memory?

```
1  double arr[] = {[11]=11, [5]=1};
```

A. 11
B. 16
C. 96
D. 88

Coding (25 pts)

21. Implement a function named max_1d to return the maximum integer value stored in a 1-d array of integers, the size of input arrays to this function is guaranteed to be greater than or equal to 1. [**10 pts**]

The max_1d function is taking two input arguments, the first argument specifies the size of the input array, the second argument is the array. Example of function calls to the max_1d function is given below:

```c
int foo[5] = {1,2,3};
printf("max_1d(5, foo) = %d\n", max_1d(5, foo)); // -> "max_1d(5, foo) = 3"
```

write the code implementation of the above max_1d() function here:

22. Given the Euclidean Algorithm to find the greatest common divisor of two integers a and b. Implement this algorithm with recursion, input arguments for a and b are **guaranteed to be positive integers**. [**15 pts**]

```
The Euclidean Algorithm: gcd(a,b)
If a is evenly divided by b (a % b == 0), // base case
    i.e., b is greatest common divisor of a and b,
    then we can return b as output.
Otherwise,  // recursive case
    Decompose a in quotient-remainder form: a = b*q + r
    Find gcd(a,b) via the recursive relationship: gcd(a,b) = gcd(b,r)
    Therefore make a recursive call gcd(b,r),
    and return gcd(b, r)'s return value as output.
```

Example of function calls to gcd function is given below:

```
printf("gcd(3, 4) = %d\n", gcd(3,4)); // gcd(3,4) = 1
printf("gcd(12, 4) = %d\n", gcd(12,4)); // gcd(12, 4) = 4
printf("gcd(35, 14) = %d\n", gcd(35,14)); // gcd(35, 14) = 7
```

write the code implementation of the above gcd() function here: