Full name: ___Solution_____
Student ID #: _____

CS 161 Midterm Exam 2, Winter 2020

1. Please put your **full name and ID number** on the top right. Ensure they are readable.
2. No devices, calculators, notes, books, Internet access, or collaboration are permitted.
3. If you need scratch paper, raise your hand and we will bring it to you. You must turn in any scratch paper you use (it will not contribute to your grade).
4. **Enter your name, ID number, and form number 1 on the Scantron.**
5. **Leave the section number blank.**
6. **Use a #2 pencil to fill in the Scantron.**

I affirm that:

(1) My answers on this exam are my own original work, without assistance from other students.
(2) I have not given, nor will I give, assistance to other students for this exam.
(3) I will not look at nor copy from other students' exams.

_____

(Sign here to agree to the above statement. **Unsigned exams will not be graded.**)

You have 50 minutes to finish the exam.

***** Good luck! *****

Part I: True/False questions are worth 2 points; Multiple-choice questions are worth 3 points.

1. A: (A: True, B: False) Every "new" statement should be followed somewhere by a "delete" statement.

2. I want to store the number of guests who've responded to my graduation party invitation. The banquet hall seats 400 people. Which of these variable types would be the best choice?
   A. `short`
   B. `unsigned short **`
   C. `int`
   D. `unsigned int <- partial credit (2/3 points)`

3. What is the output of the following C++ statement:
```
cout << 14 + 10 / 4 - 2 << endl;
```
   A. 4
   B. 12
   C. 14 **
   D. 15

4. B: (A: True, B: False) A static 2D array in C++ is laid out in column-major order.

5. What is the smallest value that can be stored in a **signed short**?
   A. -2^{15} **
   B. $-2^{15}-1$
   C. -2^{16}
   D. $-2^{16}-1$

6. What is the value of **z** after the loop ends?
```
int z;
for (z=10; z<15; z+=3)
  cout << z << endl;
```
   A. 10
   B. 13 <- Note that z gets updated at the end of each iteration, before the next z<15 check
   C. 15
   D. 16 **

7. A: (A: True, B: False) A reference allows you to access or modify the value of another variable.

8. A: (A: True, B: False) Given an array defined as
```
float dewpoint[3];
```
   and a function with prototype
```
void report(float* data);
```
   it is valid C++ to call
```
report(dewpoint);
```

9. B: (A: True, B: False) If a C++ program compiles successfully, it will have no memory leaks.

10. A: (A: True, B: False) A C-style string is a character array that includes a null terminator ('\0' character) to mark the end of the string.

11. What is the return type of `fun()`, given how it was called?

```
float rose;
double* d;
int arr = new int[3];
d = fun(3, arr);
```

A. `int`
B. `int*`
C. `double`
D. `double* **`
E. `void`

12. B: (A: True, B: False) You can have the following two function prototypes in the same program:

```
void fun(int a, int b, int c);
int* fun(int a, int b, int c);
```

13. Given this declaration:

```
int* leaf = new int[3];
```

Which valid C++ statement will store the address of a pointer?

A. `int* tree = leaf;`
B. `int* tree = &leaf;  <- &leaf is an int**, so this won't work`
C. `int** tree = leaf;`
D. `int** tree = &leaf; **`

14. B: (A: True, B: False) A dangling pointer is the result of memory not being freed.

15. A: (A: True, B: False) This array contains 6 characters:

```
char label[] = { "Super" };
```

16. Select the option that generates a random number between 7 and 14 (inclusive).

A. `rand()%7 + 14`
B. `rand()%14 + 7`
C. `rand()%8 + 7 **`
D. `rand()%7 + 8`

17. A: (A: True, B: False) `cin.getline(name, 10);` reads in 9 chars and adds '\0' to the end of the array.

18. B: (A: True, B: False) To free the memory in a 2D array, these options are equivalent:

```
(1)   for (int i=0; i<nrows; i++)
        delete [] arr[i];
      delete [] arr;
```

```
(2)   delete [] arr;
      for (int i=0; i<nrows; i++)
        delete [] arr[i];
```

19. Where is memory allocated for the right-hand side of the following statement:

```
int* holes = new int[hobbit];
```

A. Stack
B. Heap **
C. No new memory is allocated
D. Invalid C++ code <- also full credit since "hobbit" was not defined here.

20. Given the declaration `int hobbit = 10;` where is memory allocated for the right-hand side of the following statement:

```
int* wizard = &hobbit;
```

A. Stack
B. Heap
C. No new memory is allocated **
D. Invalid C++ code

21. After the following code executes, what is the value of `rose` if the user enters **-2.1**?

```
float rose;
cin >> rose;
if (rose < 0.3)
  rose += 0.5;
else if (rose < 0)
  rose -= 0.5;
else
  rose = -rose;
```

A. -2.6
B. -2.1
C. -1.6 **
D. 2.1

22. How many times does this loop iterate?

```
int q = 3;
while (q < 7)
  cout << q++ << endl;
```

A. 3
B. 4 **
C. 5
D. 7

23. Which of the following statements contain invalid C++ code?

A. `double arr[5];`
B. `int s = new int; **`
C.
```
char c[5];
char* ptr = c;
```
D. None of these are invalid.

24. What does this code segment print:

```
int list[5] = {};
*(list + 2) = 500;
cout << list[2] << "," << list << endl;
```

A. 0, 500
B. 500, 0 0 500 0 0
C. 500, a memory address **
D. a memory address, 500
E. 500, 0

25. What does this code segment print:

```
short mouse = -5;
short* cat = &mouse;
(*cat)--;
cout << mouse << endl;
```

A. -6 **
B. -5
C. -4
D. a memory address
E. error: will not compile

26. Choose the best replacement for the blank to allow this code fragment to compute the minimum value of an array called **bestsellers**.

```
short bestsellers[] = {5, 7, 10, 3, 14};
int min_index = 0;
for (int i=1; i<5; i++)
   if (bestsellers[i] < bestsellers[min_index])
            _____________________ ;
cout << "Min bestsellers: " << bestsellers[min_index] << endl;
```

A. `i = min_index;`
B. `min_index = i; **`
C. `bestsellers[i] = bestsellers[min_index]; <- overwrites contents`
D. `bestsellers[min_index] = i;`

27. How do you get the length of a C-style string called "galaxy"?

A. `galaxy.length();`
B. `len(galaxy);`
C. `galaxy.len();`
D. `galaxy.strlen();`
E. `strlen(galaxy);  **`

28. Given an array declared as `double files[7][2];` which of the following function declarations will **not** work with this array?

A. `int sort(double a[7][2]);`
B. `int sort(double a[][2]);`
C. `int sort(double a[7][]); **`

29. Given this code fragment:

```
int matrix[2][3];
int k = 3;
for (int i=0; i < 2; i++)
  for (int j=0; j < 3; j++)
       matrix[i][j] = k++;
```

The value of `matrix[1][2]` is:

A. 2
B. 3
C. 5
D. 8 **

30. B: (A: True, B: False) The dereference operator * takes precedence over the indexing operator [] .

31. What will this code print out:

```
int n_kids = 7, n_people = 5;
if (n_people > 3) {
   int n_kids = n_people * 2;
   n_people -= 2;
}
cout << n_people << ", ";
cout << n_kids << endl;
```

A. 3, 7 **
B. 3, 10 <- check scope of n_kids. Which variable are we printing?
C. 5, 7
D. 7, 3

Part II: Short Answer. (19 pts)

32. (2 pts) Let `sunburn`, `sun`, and `burn` be three Boolean variables. Write a C++ assignment statement that will set `sunburn` to true if either `sun` or `burn` is true (no conditional statement).

```
sunburn = sun || burn;
```

33. (4 pts) What is the output of the following code?

```
void update(float& p, float* q, float r) {
   p = r * 2.5;
   (*q)++;
   r = *q + p;
}

int main() {
   float hop = 5, skip = 3, jump = 1;
   update(hop, &skip, jump);
   cout << hop << "  " << skip << "  " << jump << endl;
   return 0;
}
```

```
2.54 1
```

34. (4 pts) What is the output of the following code?

```
short score[2][3];

for (int i=0; i<2; i++)
   for (int j=0; j<3; j++)
      score[i][j] = (i + j + 3) * 2;

for (int i=0; i<2; i++) {
   for (int j=0; j<3; j++)
      cout << score[i][j] - j << " ";
   cout << endl;
}
```

```
6 7 8
8 9 10
```

35. (6 pts) Fill in the blanks with valid C++ to achieve the goals indicated in the comments.

```
int peter = 7;

int& paul = peter; /* create a reference to peter */

int* mary = &peter; /* create a pointer to peter */

*mary = -7;         /* use mary to change peter to -7 */

paul++;             /* use paul to increment peter */
```

36. (4 pts) Write down a **question (and its answer)** that you think would be an appropriate addition to this test. Questions will be judged by their **relevance** to course content and **non-triviality**. Answers will be graded by their **accuracy.**

Question:

Answer:

Part III: Extra Credit. (+4 pts possible)

37. (up to 2 pts) What will the following code output?

```
float p[] = {1.1, 2.1, 3.2, 4.5};
float* q = &(p[3]) - 1;
cout << q[0] << ","
     << q[1] << endl;
```

```
3.2,4.5
```

38. (up to 2 pts) What positive input(s) would give an output of 5?

```
int input;
cin >> input;
if (input < 10 && input % 3 == 1)
  cout << input - 2 << endl;
else
     cout << input + 2 << endl;
```

```
3 or 7
```

1