

Name:										
EmpID:										

1 (30 points) Answer the following questions.

- (1) Given `int arr[] = {3, 7, 2}`, what is `arr[1]`?

Answer: `arr[1]` is 7.

- (2) Define function header for function **sort**, for two strings `a` and `b`, if `a` has more characters than `b`, swap them. Return type is `void`.

Answer: `void sort(string& a, string& b);`

- (3) Write a statement to generate a random **floating point number** in `[0, 2]`. No need to include libraries. Hint: you may use `rand` function, which returns a random integer in `[0, RAND_MAX]`.

Answer: `2.0 * rand() / RAND_MAX`

- (4) Given `string greeting = "Glad to meet you"`; What is the value for `greeting.substr(2, 6)`?

Answer: `"lad to"`

- (5) What is the value of `1 + 89 / 10`?

Answer: 9

- (6) What is the value of `foo(5)`?

```
int foo(int n) {  
    int sum = 0;  
    for (int i = 1; i <= n; i += 2)  
        sum += i;  
  
    return sum;  
}
```

Answer: 9

- (7) Suppose double variables `a` and `b` are properly declared and initialized. Declare a double variable `c` and initialize it to be $\frac{\sqrt{b}}{2a}$. You may use `sqrt` function, see cheat sheet.

Answer: `double c = sqrt(b)/(2 * a);`

- (8) What is the output of the following code?

```

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int count = 0;
6      for (int i = 10; i > 0; i -= 3)
7          count++;
8
9      cout << count << endl;
10     return 0;
11 }

```

Answer: 4

(9) What is the output of the following code?

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  void foo(int arr[], int size);
6  int main() {
7      int arr[] = {3, 1, 2};
8      int size = sizeof(arr) / sizeof(arr[0]);
9      foo(arr, size);
10
11     for (int i = 0; i < size; i++)
12         cout << arr[i] << " ";
13
14     return 0;
15 }
16
17 void foo(int arr[], int size) {
18     for (int j = size-1; j >= 1; j--)
19         for (int i = 0; i < j; i++)
20             if (arr[i] < arr[i+1])
21                 swap(arr[i], arr[i+1]);
22 }

```

Answer: 3 2 1

(10) Write a condition to represent that integer i is larger than 0 **and** i is even. An integer is even if it can be divided by 2.

Answer:

`(i > 0 && i % 2 == 0)`

or

```
(i > 0 and i % 2 == 0)
```

2 (20 points) Answer the following questions.

(1) What is the output of the following code?

```
1  #include <iostream>
2  using namespace std;
3
4  void show(int size);
5  int main() {
6      show(4);
7      return 0;
8  }
9
10 void show(int size) {
11     for (int row = 0; row < size; row++) {
12         for (int col = 0; col < size; col++)
13             if (row + col == size-1)
14                 cout << "*";
15             else cout << "-";
16
17         cout << endl;
18     }
19 }
```

Answer:

```
---*
--*-
-*--
*---
```

(2) Define function `no_digit`, for a string, return true if **none** of its elements is a digit character, that is, one from '0' to '9', otherwise, return false. For example, `no_digit("abc")` returns true and `no_digit("ab1")` returns false. **No need to include libraries or define main function.**

Hint: you may use `int isdigit(int ch)` to test whether a character is digit letter '0' - '9' or not.

Answer:

```
1  #include <iostream>
2  #include <string>
3  #include <cctype>
4
5  using namespace std;
6
```

```
7  bool no_digit(string str);
8
9  int main() {
10     cout << boolalpha << no_digit("ab1") << endl; //false
11     cout << boolalpha << no_digit("abc") << endl; //true
12     return 0;
13 }
14
15 bool no_digit(string str) {
16     for (int i = 0; i < str.length(); i++)
17         if ( isdigit(str[i]) )
18             return false;
19
20     return true;
21 }
```

3 (50 points) Programming exercises

- (1) Define function called `percentage`, for an array of ints, its size, and a target, return the percentage of integers that is **larger than** the target to the size of the array.

For example, if the array has elements 3, 2, -1, and target is 1, then there are 2 integers out of 3 elements that is larger than the target, so the percentage is 66.6667%. The return is 66.6667, a floating point number.

Answer:

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  double percentage(int arr[], int size, int target);
6
7  int main() {
8      int arr[] = {3, 2, -1};
9      int size = sizeof(arr) / sizeof(arr[0]);
10     int target = 1;
11     cout << percentage(arr, size, target) << "%" << endl;
12     return 0;
13 }
14
15 double percentage(int arr[], int size, int target) {
16     int numPositives = 0;
17     for (int i = 0; i < size; i++)
18         if (arr[i] > target)
19             numPositives++;
20
21     return 100.0 * numPositives / size;
22 }
```

In main function, declare array `arr` with values 3, 2, -1. Call the above function on `arr` with appropriate size and target 1. Print out the return. **Just write the statements in main function, no need to include libraries.**

Answer:

```
1  int arr[] = {3, 2, -1};
2  int size = sizeof(arr) / sizeof(arr[0]);
3  int target = 1;
4  cout << percentage(arr, size, target) << "%" << endl;
```

- (2) Write code in main to enter a series of integers until -1 is entered. Find out the number of integers that can be divided by both 3 and 5 at the same time, the number of integers that can be divided by 3 only, and the number of integers that can be divided by 5 only.

Integer num is divided by 3 means the remainder of num divided by 3 is 0.

```
Enter an int (-1 to stop): 5
Enter an int (-1 to stop): 3
Enter an int (-1 to stop): 15
Enter an int (-1 to stop): 7
Enter an int (-1 to stop): 6
Enter an int (-1 to stop): 30
Enter an int (-1 to stop): -1
number can be divided by 3 and 5: 2
number can be divided by 3 only: 2
number can be divided by 5 only: 1
```

Answer:

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  //sample input/output:
6  //Enter an int (-1 to stop): 5
7  //Enter an int (-1 to stop): 3
8  //Enter an int (-1 to stop): 15
9  //Enter an int (-1 to stop): 7
10 //Enter an int (-1 to stop): 6
11 //Enter an int (-1 to stop): 30
12 //Enter an int (-1 to stop): -1
13 //number can be divided by 3 and 5: 2
14 //number can be divided by 3 only: 2
15 //number can be divided by 5 only: 1
16
17 int main() {
18     cout << "Enter an int (-1 to stop): ";
19     int elm;
20     cin >> elm;
21
22     int multiple_15 = 0; //divided by 3 and 5
23     int multiple_3 = 0; //divided by 3
24     int multiple_5 = 0; //divided by 5
25     while (elm != -1) {
26         if (elm % 3 == 0 && elm % 5 == 0)
27             multiple_15++;
28         else if (elm % 3 == 0)
29             multiple_3++;
30         else if (elm % 5 == 0)
```

```
31         multiple_5++;
32
33         cout << "Enter an int (-1 to stop): ";
34         cin >> elm;
35     }
36
37     cout << "number can be divided by 3 and 5: " << multiple_15 << endl;
38     cout << "number can be divided by 3 only: " << multiple_3 << endl;
39     cout << "number can be divided by 5 only: " << multiple_5 << endl;
40     return 0;
41 }
```