

## Test 1 review:

Look over: program examples discussed in class (class notes), quiz, closed labs, and open lab code.

### Topics:

1. Evaluate expression involving implicit/explicit type cast, and +, -, \*, /, % operators
2. Write C++ expression based on math formula given (e.g., compute the roots of quadratic equation, compute the distance between two points on Euclidean space)
  - o Math library functions
3. Be able to write C++ statement for formatted output based on description
  - o fixed, scientific, showpoint, setprecision, left, right, setw
4. Evaluate logical expression involving ==, !=, !, <, <=, >, >=, &&, || operators
5. Write logical expression based on description
6. Be able to evaluate C++ code with while loop
7. Be able to evaluate C++ code with if/else statement, simple if statement, if/else if/else statement
8. Be able to write a complete C++ program involving
  - o Preprocessors, Constant definition, Variable declaration, Prompt user for input
  - o Decision statement, Repetition statement, Display output on screen
9. String functions
  - o getline(), strlen(), find(), substr()
  - o Be able to write program using string functions
10. File operation
  - o open, close, assert
  - o read from and write to file
  - o be able to read information from a datafile and process the information.
11. Define functions (declaration, activation, definition)
  - o void function vs. value returning functions
    - (a) be able to write functions for different problems
  - o Parameter passing (value parameter vs. reference parameter)
  - o Be able to trace program with functions and parameters
12. for loop and do-while loop
13. One dimensional array
  - o Using for loop with one dimensional array
  - o Linear search and binary search
  - o Sorting
  - o Adding and deleting values in array

### Example questions:

1. a. The value of the C++ expression:  $7 / 4 * 5$  is:
  - A) 7
  - B) 5.0
  - C) 5
  - D) 2.45
  - E) None of the above
- b. The value of the C++ expression:  $2 + 26 \% 5 - 3$  is:
  - A) 6
  - B) 4
  - C) 5
  - D) 4.2
  - E) none of the above

- f. If x is a float variable containing a positive value, which of the following statements outputs the value of x, **rounded to the nearest tenth** digit of the decimal?
- A) `cout << int(x*10) + 0.5;`
  - B) `cout << int(x*10 + 0.5)/10.0;`
  - C) `cout << int(x*10)/10;`
  - D) `cout << float(x + 0.05);`
  - E) `cout << x + int(0.05);`
- i. If p is a Boolean variable, which of the following logical expressions always has the value **true**?
- A) `p && p`
  - B) `p || p`
  - C) `p && !p`
  - D) `p || !p`
  - E) b and d above

2. (a) Write a C++ assignment statement that implements the following formula, assuming result, p1, p2, x1, x2, y1, and y2 are variables of float type

$$result = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$result = \sqrt{a^2 + b^2}$$

$$result = a^3 - 3a^2b + 3ab^2 - b^3$$

(b) Calculate this expression:

$$\text{pow}(2.0, 2.0) - \text{sqrt}(\text{int}(121.5)) - 13 * \text{int}(19.5 - 8.5) * 2 / 3 \% 4$$

3. Given the input data:

25 10 6 -1 7

what is the output of the following code fragment? (All variables are of type int.)

```
sum = 0;
cin >> number;
while (number != -1)
{
    cin >> number;
    sum = sum + number;
    cout << number << '\t' << sum << endl;
}
```

5.(a) What is the output of the following two code fragments?

```
n = 0;
while (n <= 10) {
    cout << n << ' ';
    n=n+2;
}
n=1;
sum=0;
while (n < 20)
{
    cout << n << ' ' << sum << endl;
    sum = sum + n;
```

```

        n=n*2;
    }

```

(b) After execution of the following code, what is the value of length? (both variables: count and length are of type int.)

```

length = 50;
count = 4;
while (count <= 5)
{
    if (length >= 100)
        length = length - 2;
    else
        length = count * length;

    count = count + 1;
}

```

6.

Given the input data: 25 100 6 -1 17

what is the output of the following code fragment? (All variables are of type int.)

```

sum = 0;
cin >> number;
while (number != -1)
{
    cin >> number;
    sum = sum + number;
    cout << number << '\t' << sum << endl;
}

```

7. Write a complete C++ program that prompts the user to enter an integer value, then the program computes and displays all the divisors of this integer. Divisor is an integer that divides an integer evenly. For example 12/4 is 3, so 4 is a divisor for 12; another example: 160 divides 40 is 4, so 40 is a divisor for 160.

8. Write a complete C++ program that will read integer values from a data file. This program will display the total number of values read from the data file, the largest number read, the smallest number read, and the average of all the values read.

9. Write a **complete C++ program** for the following problem.

Prompt the user to enter the radius of a sphere. Then compute and display the surface area and volume of the sphere. The formulae for the computation are:

$$surface\ area = 4\pi R^2, \quad volume = \frac{4}{3}\pi R^3$$

where R is the radius of the sphere.

You are required to write the following three user defined functions for this program:

- **GetRadius:** This function prompts the user to enter the radius value. It checks to see if the radius value is valid, i.e., it is greater than 0. It continues to prompt the user until a valid radius value is entered. It then returns the radius value to the calling function.
- **ComputeStatistics:** This function computes the **surface area** and the **volume** based on the radius of a sphere. The computed values are sent back to the calling function with reference parameters.
- **DisplayResults:** This function displays the radius value, the surface area, and the volume of a sphere.

Make sure to include all the function declarations and definitions. Think carefully whether to use void function or value returning function, and for parameter passing, whether to use value parameter or reference parameter.

10. Write a **C++ program** to count and display the number of occurrence of the word “**to**” in a line of text entered by user.

*Example Run:*

*Enter Text: to be or not to be, that is going to be the hard question.*      ← user input

*The word appeared 3 times in this sentence.*      ← program output

11. Write a complete C++ program to read a list of movie titles from a data file named “**movies.dat**”. The movie titles are written one title per line in the data file. Your program reads the movie titles and displays them one by one on screen. The capacity of the array is set to 100. The program reads the movie titles til the end of the data file is reached or a maximum of 100 values is read. The program then displays a message showing the number of movie titles read from the data file.

### Function related problems:

1. Multiple choice questions:

- 1) Given the function prototype

void FixThis(char&, int&, float, int);

which of the following is an appropriate function call? (someChar is of type char, someInt is of type int, and someFloat is of type float.)

- A) FixThis('o', someInt, 6.85, int('c'));
- B) someFloat = 0.3 \* FixThis(someChar, someInt, someInt, someFloat);
- C) FixThis('p', someInt + 5, someFloat, 50);
- D) FixThis(someChar, someInt, someFloat, int('c'));
- E) FixThis(someChar, someInt, someFloat);

- 2) Consider the function definition

```
void DoThis( int & alpha,  int & beta )
{
    int temp;

    temp=50;
    temp = alpha;
    alpha = beta;
    beta = temp;
    return;
}
```

Suppose that the caller has integer variables **gamma** and **delta** whose values are 10 and 20, respectively. What are the values of gamma and delta after the following function call?

DoThis(gamma, delta);

- A) gamma = 50 and delta = 20
- B) gamma = 20 and delta = 10
- C) gamma = 50 and delta = 10
- D) gamma = 10 and delta = 20
- E) gamma = 20 and delta = 20

3) Consider the function definition

```
void Demo( int  value1,  float& value2 )
{
    value1 = value1* 3;
    value2 = int(value2) + 5.5;
    return;
}
```

Suppose that the caller has variables **myInt** and **myFloat** whose values are 5 and 2.5, respectively. What are the values of myInt and myFloat after return from the following function call?

Demo(myInt, myFloat);

- A) myInt = 15 and myFloat = 5.5
- B) myInt = 15 and myFloat = 7.5
- C) myInt = 5 and myFloat = 5.5
- D) myInt = 5 and myFloat = 7.5
- E) none of the above

4) Given the function definition:

```
void Twist( int  a,  int& b )
{
    int  c;

    a = b + 2;
    c = a * 3;
    b = c + a;
}
```

What is the output of the following code fragment that calls function Twist? (All variables are of type int.)

```
r = 1;
s = 2;
t = 3;
Twist(s, t);
cout << r << ' ' << s << ' ' << t << endl;
```

2. Given the function declaration for “Processing” and local variables declared in the main function:

```
#include <iostream>
using namespace std;
void Processing(float &, float, int &, int, char);
int main ()
{
    // local variable declaration
    int id, socialSecurity;
    float salary, tax;
    char init;

    // the rest of the program omitted
    .....
}
```

```
void Processing(float & n1,
               float n2,
               int & sSN,
               int id,
               char finitial)
{
    // content of function omitted
    .....
}
```

- (a) Which parameters are passed by value? Which parameters are passed by reference?
- (b) State whether each of the following four calls of the function “Processing” in the main function is correct or incorrect. If it is incorrect, briefly explain why.

- (1) Processing(35.5, 0.0085, socialSecurity, id, 'F');
- (2) Processing(salary, tax, socialSecurity, 41);
- (3) Processing(salary\*0.15, tax\*0.2, socialSecurity, id, 'L');
- (4) Processing(salary, tax, socialSecurity, 45.5, 'L');

3. Show the output of the following program:

```
#include <iostream>
using namespace std;
void Test (int&, int);

int main ( )
{
    int d=12;
    int e=14;

    Test(d, e);
    cout << "In the main function after the first call, d=" << d << ", e=" << e << endl;

    d=15;
    e=18;
    Test (e, d);
    cout << "In the main function after the second call, d=" << d << ", e=" << e << endl;

    return 0;
}

void Test(int &s, int t)
{
    s=5;
    s = s+10;
    t = 4*s;
    cout << "in function Test, s=" << s << " " << "t=" << t << endl;

    return;
}
```

4. Write a *value returning function* named **ComputePostage** that returns the cost of mailing a package, given the weight of the package in pounds and ounces, and the cost per ounce. (1 pound = 16 ounces) Fill in the following incomplete program:
- the declaration of the function
  - call to function **ComputePostage** that computes the postage for a package
  - Show the definition of the function

// fill in function declaration here

```
int main( )
{
    int pounds, ounces;
    float costPerOunce;
    float cost;

    cout << "How much does the package weight? (enter pounds and ounces)" << endl;
    cin >> pounds >> ounces;

    cout << "What is the cost of the package per ounce? " << endl;
    cin >> costPerOunce;

    // fill in function call here

    _____
    cout << "This package costs $" << cost << "." << endl;

    return 0;
}
// fill in function definition here
```

### One dimensional array related questions:

5. Given the array declared and initialized as the following:
- ```
const int SIZE=8;
int values[SIZE] = {2, 11, 4, 63, 66, 25, 35, 88};
```
- Show the C++ statements needed to compute and display the average of all the values in the array “values”.
6. What is the output of the following program?
- ```
const int SIZE=8;
int values[SIZE] = {2, 11, 4, 63, 66, 25, 35, 88};
int index;
index=0;
for (int j=1; j<SIZE; j++) {
    if (values[j] < values[index]) {
        index = j;
    }
}
cout << index << “ “ << values[index] << endl;
```



7. Given the following array declaration, show the C++ statements needed to read a maximum of 100 values from the data file. Or if the data file contains less than 100 values, it reads all the values in the file. It stores all the values in the array and displays how many value were read.

```
const int SIZE = 100;
int array[SIZE];
ifstream myIn; myIn.open("values.dat"); assert(myIn);
```

8. Write a complete C++ program to:
- Read float type values from a data file named "values.dat" into a one-dimensional array named "values" (the maximum capacity of the array is set to 100). The values are stored one value per line in the data file. There are an unknown number of values in the data file. Your program reads the values til it reaches the end of the data file or the maximum capacity of the array is reached.

After the values are read and stored in the array:

- Sort the values in ascending order. Write a **Sort function** for this task. Call the function in the main function.
- Find and return the array subscript corresponding to the minimum value in this array. Write a **FindMinIndex** function for this task. Call the function in the main function and display that array subscript.

For (b) and (c), you need to show the complete function declaration, activation and definitions

9. Add a user defined function **InsertAtFront** to Question 1 such that a user may add a new value to those currently stored in an array of float type values.
- If the array is already full, display a message to the user stating that insertion can not be carried out.
  - If the array still has room for more values, then prompt the user to enter a new value, and store that value **at the beginning of the array, i.e., at array location having subscript 0. This means all the values that were in the array need to be shifted to make room for this new value.**

Show the complete C++ function definition for the function **InsertAtFront**.

10. Given two **parallel** one dimensional arrays in the main function as shown below: **airportCode** and **airportName**. **airportCode** contains the codes for major airports, for example, BNA is the code for the Nashville International Airport and ATL is the code for the Atlanta Airport; and **airportName** contains the names of the airports. The total number of airport codes/names is stored in variable "numOfAirports".

<b>airportCode</b>	<b>airportName</b>
BNA	Nashville Airport
ATL	Atlanta Airport
SFO	San Francisco Airport
SJC	San Jose Airport
JFK	J F Kennedy New York Airport
ORD	Chicago O'Hare Airport
LAX	Los Angeles Airport
...	...
...	...
DCA	Washington DC – National Airport

Write a C++ **user defined function** named **FindAirportName** that prompts the user to enter an airport code, and displays the corresponding airport name. The input parameters for this function include: the array **airportCode**, the array **airportNames**, and **the number of items stored in these arrays**

11. **This problem has two parts. (The second part is on the back page)**

The function **Insert** does not work quite the way it is supposed to.

- a. Step through the code shown below and write down the output of the current version of the program.

```
#include <iostream>
using namespace std;
```

```
int Insert(int [], int, int, int);
const int SIZE = 10; // maximum number of items to store in array
```

```
int main()
{
    int array[SIZE], value=0, position=1, aSize=4;

    for (int i=0; i<aSize; i++) // initialize array
        array[i] = 2*i+1;

    // insert value into array at position
    aSize = Insert(array, value, position, aSize);

    // display values after the insertion
    for (int i=0; i<aSize; i++)
        cout << array[i] << " ";

    return 0;
}
```

Show program output here:

```
// this function inserts "element" in the given "position" in array "arr". It returns the new array size
int Insert(int arr[], int element, int position, int size)
{
    for (int i=position; i<size; i++)
        arr[i+1] = arr[i];
    arr[position]= element;
    size=size+1;

    return size;
}
```

- b. How would you modify the function Insert to correctly insert an element into the array at position "position"? For example, before insertion, array looks like this: 5 7 9 2. After the insertion of element 6 at position 2, array looks like this: 5 7 6 9 2