E.S MARIE Adelaide

Computer science department

S5MCE

COVID -19 PERIOD EXERCISES [STAY@HOME]

A. C++PROGRAMMING

I. Elementary c++ programming

1. Describe the two ways to include comments in a C++ program
2. What is wrong with this program?
   ```
   #include <iostream>
   int main()
   { // prints "Hello,World!":
   cout << "Hello,World!\n"
   }
   ```
3. What is wrong with the following C-style comment?
   ```
   cout << "Hello,/* change? */ World.\n";
   ```
4. What's wrong with this program:
   ```
   #include <iostream>;
   int main
   { // prints "n = 22":
   n = 22;
   cout << "n = << n << endl;
   }
   ```
5. What does a declaration do?
6. What is the purpose of the preprocessing directive:
   ```
   #include <iostream>
   ```
7. What is the shortest possible C++ program?
8. Where does the name "C++" come from?
9. What's wrong with these declarations:
   ```
   int first = 22,last = 99,new = 44,old = 66;
   ```
10. In each of the following, assume that m has the value 5 and n has the value 2 before the statement executes. Tell what the values of m and n will be after each of the following statements executes:
    a. m *= n++;
    b. m += --n;

11. Evaluate each of the following expressions, assuming in each case that m has the value 25
    and n has the value 7:
    a. m - 8 - n
    b. m = n = 3
    c. m%n
    d. m%n++
    e. m%++n
    f. ++m - n—

12. Parse the following program, identifying all the keywords, identifiers, operators, literals, punctuation, and comments:

```
int main()
{ int n;
cin >> n;
n *= 3; // multiply n by 3
cout << "n=" << n << endl;
}
```

13. Identify and correct the error in each of the following:
    a. cout >> count;
    b. int double=44;
14. How do the following two statements differ:
    char ch = 'A';
    char ch = 65;
15. What code could you execute to find the character whose ASCII code is 100?
16. What does “floating-point” mean, and why is it called that?
17. What is numeric overflow?
18. How is integer overflow different from floating-point overflow?
19. What is a run-time error? Give examples of two different kinds of run-time errors.
20. What is a compile-time error? Give examples of two different kinds of compile-time errors.
21. Write four different C++ statements, each subtracting 1 from the integer variable n
22. Write a block of C++ code that has the same effect as the statement
    n = 100 + m++;
23. Write a block of C++ code that has the same effect as the statement
    n = 100 + ++m;
    without using the pre-increment operator
24. Write a single C++ statement that subtracts the sum of x and y from z and then increments y.
25. Write a single C++ statement that decrements the variable n and then adds it to total.
26. Write a program that prints the first sentence of the Gettysburg Address (or your favorite quotation)
27. Write a program that prints the block letter “B” in a 7 × 6 grid of stars like this:

```
*****
*    *
*    *
*****
*    *
*    *
*****
```

28. Write and run a program that prints the first letter of your last name as a block letter in a 7 × 7 grid of stars.

29. Write and run a program that shows what happens when each of the following ten "escape sequences" is printed: \a, \b, \n, \r, \t, \v, \', \", \\, \?.

30. Write and run a program that prints the sum, difference, product, quotient, and remainder of two integers. Initialize the integers with the values 60 and 7.

31. Write and run a program that prints the sum, difference, product, quotient, and remainder of two integers that are input interactively.

32. Write and run a test program that shows how your system handles uninitialized variables.

33. Write and run a program that causes negative overflow of a variable of type short

34. Write and run a program that demonstrates round-off error by executing the following steps:
(1) initialize a variable a of type float with the value 666666; (2) initialize a variable b of type float with the value 1-1/a; (3) initialize a variable c of type float with the value 1/b - 1; (4) initialize a variable d of type float with the value 1/c + 1; (5) print all four variables. Show algebraically that d = a even though the computed value of d ≠ a. This is caused by round-off error.

II. SELECTION

1. Write a single C++ statement that prints "Too many" if the variable count exceeds 100.
2. What is wrong with the following code:

a. cin << count;
b. if x < y min = x
else min = y;

3. What is wrong with this code:

```
cout << "Enter n: ";
cin >> n;
if (n < 0)
cout << "That is negative. Try again." << endl;
cin >> n;
        else
        cout << "o.k. n = " << n << endl;
```

4. What is the difference between a reserved word and a standard identifier?
5. State whether each of the following is true or false. If false, tell why.
   a. !(p || q) is the same as !p || !q
   b. !!!p is the same as !p
   c. p && q || r is the same as p && (q || r)
6. Construct a truth table for each of the following boolean expressions, showing its truth value (0 or 1) for all 4 combinations of truth values of its operands p and q.
   a. !p || q
   b. p&&q || !p&&!q
   c. (p||q) && !(p&&q)

7. Use truth tables to determine whether the two boolean expressions in each of the following
   are equivalent.
   a. !(p && q) and !p && !q
   b. !!p and p
   c. !p || q and p || !q
   d. p && (q && r) and (p && q) && r
   e. p || (q && r) and (p || q) && r
8. What is short-circuiting and how is it helpful?
9. What is wrong with this code:
   if (x = 0) cout << x << " = 0\n";
   else cout << x << " != 0\n";
10. What is wrong with this code:
    if (x < y < z) cout << x << " < " << y << " < " << z << endl;
11. Construct a logical expression to represent each of the following conditions:
    a. score is greater than or equal to 80 but less than 90;
    b. answer is either 'N' or 'n';
    c. n is even but not 8;
    d. ch is a capital letter.
12. Construct a logical expression to represent each of the following conditions:
    a. n is between 0 and 7 but not equal to 3;
    b. n is between 0 and 7 but not even;
    c. n is divisible by 3 but not by 30;
    d. ch is a lowercase or uppercase letter
13. What is wrong with this code:
    if (x == 0)
    if (y == 0) cout << "x and y are both zero." << endl;
    else cout << "x is not zero." << endl;
14. What is the difference between the following two statements:
    if (n > 2) { if (n < 6) cout << "OK"; } else cout << "NG";
    if (n > 2) { if (n < 6) cout << "OK"; else cout << "NG"; }
15. What is a “fall-through?
16. How is the following expression evaluated?
    (x < y ? -1 : (x == y ? 0 : 1));
17. Write a single C++ statement that uses the conditional expression operator to assign the absolute value of x to absx.
18. Write a single C++ statement that prints “too many” if the variable count exceeds 100, using
    a. an if statement;
    b. the conditional expression operator.
19. Write and run a program that reads the user’s age and then prints “You are a child.” if the age < 18, “You are an adult.” if 18 ≤ age < 65, and “You are a senior citizen.” if age ≥ 65

20. Write and run a program that reads two integers and then uses the conditional expression operator to print either “multiple” or “not” according to whether one of the integers is a mul
    tiple of the other

21. Write and run a program that simulates a simple calculator. It reads two integers and a character. If the character is a +, the sum is printed; if it is a -, the difference is printed; if it is a *,
    the product is printed; if it is a /, the quotient is printed; and if it is a %, the remainder is printed. Use a switch statement
22. Write and test a program that solves quadratic equations. A quadratic equation is an equation
    of the form ax2 + bx + c = 0, where a, b, and c are given coefficients and x is the unknown.
    The coefficients are real number inputs, so they should be declared of type float or double. Since quadratic equations typically have two solutions, use x1 and x2 for the solutions to be output. These should be declared of type double to avoid inaccuracies from
    round-off error
23. Write and run a program that reads a six-digit integer and prints the sum of its six digits. Use
    the quotient operator / and the remainder operator % to extract the digits from the integer. For example, if n is the integer 876,543, then n/1000%10 is its thousands digit 6

III. ITERATIONS

1. What happens in a **while** loop if the control condition is false (*i.e.*, zero) initially?
2. When should the control variable in a **for** loop be declared before the loop (instead of within
   its control mechanism)?
3. How does the **break** statement provide better control of loops?
4. What is the minimum number of iterations that
   a. a while loop could make?
   b. a do..while loop could make?

5. What is wrong with the following loop:
   while (n <= 100)
   sum += n*n;

6. If s is a compound statement, and e1, e2, and e3 are expressions, then what is the difference
   between the program fragment:
   for (e1; e2; e3)
   s;
   and the fragment:
   e1;

```
while (e2)
{ s;
e3;
}
```

7. What is wrong with the following program:
```
int main()
{ const double PI;
int n;
PI = 3.14159265358979;
n = 22;
}
```
8. What is an "infinite loop," and how can it be useful?
9. How can a loop be structured so that it terminates with a statement in the middle of its block?
10. Why should tests for equality with floating-point variables be avoided?

11. Trace the following code fragment, showing the value of each variable each time it changes:

```
float x = 4.15;

for (int i=0; i < 3; i++)

x *= 2;
```

12. Assuming that e is an expression and s is a statement, convert each of the following for loops into an equivalent while loop:
a. for (; e;) s
b. for (; ; e) s

13. Convert the following for loop into a while loop:
```
for (int i=1; i <= n; i++)
cout << i*i << " ";
```

14. Describe the output from this program:
```
int main()
{ for (int i = 0; i < 8; i++)
if (i%2 == 0) cout << i + 1 << "\t";
else if (i%3 == 0) cout << i*i << "\t";
else if (i%5 == 0) cout << 2*i - 1 << "\t";
else cout << i << "\t";
}
```

15. Describe the output from this program:
```
int main()
{ for (int i=0; i < 8; i++)
{ if (i%2 == 0) cout << i + 1 << endl;
else if (i%3 == 0) continue;
```

```
else if (i%5 == 0) break;
cout << "End of program.\n";
}
cout << "End of program.\n";
}
```

16. In a 32-bit float type, 23 bits are used to store the mantissa and 8 bits are used to store the exponent.
a. How many significant digits of precision does the 32-bit float type yield?
b. What is the range of magnitude for the 32-bit float type?

17. Write and run a program that uses a while loop to compute and prints the sum of a given number of squares. For example, if 5 is input, then the program will print 55, which equals 1^2
$+ 2^2 + 3^2 + 4^2 + 5^2$

18. Write and run a program that uses a for loop to compute and prints the sum of a given number of squares

19. Write and run a program that uses a do..while loop to compute and prints the sum of a given number of squares.

20. Write and run a program that directly implements the quotient operator / and the remainder
operator % for the division of positive integers.

21. Write and run a program that reverses the digits of a given positive integer. ()

22. Apply the Babylonian Algorithm to compute the square root of 2. This algorithm (so called
because it was used by the ancient Babylonians) computes by repeatedly replacing one estimate x with the closer estimate (x + 2/x)/2. Note that this is simply the average of x and
2/x.

23. Write a program to find the integer square root of a given number. That is the largest integer
whose square is less than or equal to the given number

24. Implement the Euclidean Algorithm for finding the greatest common divisor of two given positive integers. This algorithm transforms a pair of positive integers (m, n) into a pair (d, 0)
by repeatedly dividing the larger integer by the smaller integer and replacing the larger withthe remainder. When the remainder is 0, the other integer in the pair will be the greatest common divisor of the original pair (and of all the intermediate pairs). For example, if m is 532
and n is 112, then the Euclidean Algorithm reduces the pair (532,112) to (28,0) by (532,112) → (112,84) → (84,28) → (28,0).
So 28 is the greatest common divisor of 532 and 112. This result can be verified from the facts that 532 = 28·19 and 112 = 28·8. The reason that the Euclidean Algorithm works is that
each pair in the sequence has the same set of divisors, which are precisely the factors of

the
greatest common divisor. In the example above, that common set of divisors is {1, 2, 4, 7, 14,
28}. The reason that this set of divisors is invariant under the reduction process is that when
m = n·q + r, a number is a common divisor of m and n if and only if it is a common divisor of
n and r.

IV. VISUAL BASIC DOT NET(2015)

1. By using function make a BMI application and after write all 20steps used to a setup