



**READ THIS NOW!**

**Failure to read and follow the instructions below may result in severe penalties.  
Failure to adhere to these directions will not constitute an excuse or defense.**

- Print your name in the space provided below.
- Print your name and ID number on the Opscan form; be sure to code your ID number on the Opscan form. Code **Form A** on the Opscan; and code your section group number: 12:00 = 1; 2:00 = 2; 3:00 = 3.
- Choose the single best answer for each question – some answers may be partially correct. If you mark more than one answer, it will be counted wrong.
- Unless a question involves determining whether given C++ code is syntactically correct, assume that it is. The given code has been compiled and tested, except where there are deliberate errors. Unless a question specifically deals with compiler #include directives, you should assume the necessary header files have been included.
- Note that questions about printed values disregard formatting completely.
- Be careful to distinguish integer values from floating point (real) values (containing a decimal point).. In questions/answers which require a distinction between integer and real values, integers will be represented without a decimal point, whereas real values will have a decimal point, [ 1044 (integer), 1044.0 (real)].
- When you have completed the test, sign the pledge at the bottom of this page and turn in the test and your Opscan.
- **This is a closed-book, closed-notes examination. No calculators or other electronic devices may be used during this examination. You may not discuss (in any form: written, verbal or electronic) the content of this examination with any student who has not taken it. You must return this test form when you complete the examination. Failure to adhere to any of these restrictions is an Honor Code violation.**
- There are 40, 2.5-point multiple-choice questions.
- The answers you mark on the Opscan form will be considered your official answers.

**Do not start the test until instructed to do so!**

Name \_\_\_\_\_

**Pledge:** On my honor, I have neither given nor received unauthorized aid on this examination.

\_\_\_\_\_

For the next two questions, consider the execution of the following code fragment:

```
int I = 12, J = 0;
while (I > 8){
    I = I - 1;
    J = J + I;
}
cout << "I = " << I << endl;
cout << "J = " << J << endl;
```

1) What value is printed for the variable I:

- 1) 4                      2) 5                      3) 6                      4) 7                      5) 8  
6) 9                      7) 10                      8) none of the above

2) What value is printed for the variable J:

- 1) 0                      2) 4                      3) 11                      4) 21                      5) 30  
6) 38                      7) 45                      8) none of the above
- 

3) Which of the following are iteration mechanisms in C++?

- 1) while                      2) repeat...until                      3) for  
4) if..else                      5) switch                      6) all of the above  
7) 1 and 3 only                      8) 4 and 5 only                      9) 1, 3, 4 and 5 only  
10) none of the above
- 

For the next two questions, consider executing the following code fragment (assume any additional declarations, etc, needed to make the code syntactically correct):

```
int j = 107;
double varq = 1.0, newq = 0.0;
while (j > 30) {
    newq = j + varq;
    j = j - 4;
}
```

4) How many times will the body of the loop be executed?

- 1) 107                      2) 19                      3) 79                      4) 21                      5) 20  
6) none of the above

5) What is the value of j after the last iteration of the loop?

- 1) 27                      2) 28                      3) 30                      4) 31                      5) none of the above

For the next three questions, assume the input file stream `ifile` is connected to a file containing the following data:

5        3        -2        4        1        7

Consider the execution of the code fragment given in each question and determine the value that would be printed.  
Choose from the following answers:

- 1) 3                    2) 5                    3) 6                    4) 8                    5) 10  
6) 11                   7) 13                   8) 18                   9) none of the above

6) 

```
int sum = 0, count, mystery, value;
ifile >> mystery;
for (count = 1; count <= mystery; count++) {
    ifile >> value;
    sum = sum + value;
}
cout << "sum = " << sum;
```

7) 

```
int sum = 0, mystery;
ifile >> mystery;
while (ifile) {
    sum = sum + mystery;
    ifile >> mystery;
}
cout << "sum = " << sum;
```

8) 

```
int sum = 0, mystery;
ifile >> mystery;
while (mystery > 0) {
    sum = sum + mystery;
    ifile >> mystery;
}
cout << "sum = " << sum;
```

---

For the next two questions, consider executing the following code fragment (assume any additional declarations, etc, needed to make the code syntactically correct):

```
int    j = 2;
while (j != 28) {
    cout << j << endl;
    j = j + 3;
}
```

9) Exactly how many times will the body of the loop be executed?

- 1) 8                    2) 9                    3) 10                   4) 11                   5) none of the above

10) What is the tenth value printed?

- 1) 2                    2) 14                   3) 23                   4) 26                   5) 28  
6) 29                   7) 47                   8) none of these

Consider executing the following program:

```
#include <iostream.h>
void main() {
    int i, j, sum;
    sum = 0;
    for (i = 1; i <= 3; i++) {
        for (j = 1; j <= 3; j++)
            sum++;
        cout << sum << endl;
    }
}
```

11) What is the value printed on the second line of output?

- |      |                      |      |      |      |      |
|------|----------------------|------|------|------|------|
| 1) 0 | 2) 1                 | 3) 2 | 4) 3 | 5) 4 | 6) 5 |
| 7) 6 | 8) none of the above |      |      |      |      |

Now, consider executing the following slightly different program:

```
#include <iostream.h>
void main() {
    int i, j, sum;
    for (i = 1; i <= 3; i++) {
        sum = 0;
        for (j = 1; j <= 3; j++)
            sum++;
        cout << sum << endl;
    }
}
```

12) What is the value printed on the second line of output?

- |      |                      |      |      |      |      |
|------|----------------------|------|------|------|------|
| 1) 0 | 2) 1                 | 3) 2 | 4) 3 | 5) 4 | 6) 5 |
| 7) 6 | 8) none of the above |      |      |      |      |

---

Assume the following declarations:

```
void Fix(int& , float );
int someInt = 42;
float someFloat = 3.14;
```

13) Which of the following is an appropriate call of the function Fix?

- |                                 |                       |
|---------------------------------|-----------------------|
| 1) Fix(24, 6.85);               | 6) all of the above   |
| 2) Fix(someInt, 6.85);          | 7) 1 through 5 only   |
| 3) Fix(24, someFloat);          | 8) 1 and 3 only       |
| 4) Fix(someInt, someFloat);     | 9) 2 and 4 only       |
| 5) Fix(someInt + 5, someFloat); | 10) none of the above |

14) A function parameter should be passed by reference if the parameter's data flow (communication) is:

- |  |                      |
|--|----------------------|
| 1) one-way, into the function            | 5) 1 and 2 only      |
| 2) one-way, out of the function          | 6) 1 and 3 only      |
| 3) two-way, into and out of the function | 7) 2 and 3 only      |
| 4) all of the above                      | 8) none of the above |
- 

15) A function, `someFunc`, has two formal parameters, `Alpha` and `Beta`, both of type `int`. The data flow (communication) for `Alpha` is one-way, into the function. The data flow for `Beta` is two-way, into and out of the function. Which of the following is the most appropriate prototype for `someFunc`?

- |   |  |
|---|--|
| 1) <code>someFunc(int Alpha, int Beta);</code>      | 4) <code>someFunc(int&amp; Alpha, int&amp; Beta);</code> |
| 2) <code>someFunc(int&amp; Alpha, int Beta);</code> | 5) all of the above                                      |
| 3) <code>someFunc(int Alpha, int&amp; Beta);</code> | 6) none of the above                                     |
- 

16) Which of the following statements about value parameters is true?

- |  |                      |
|--|----------------------|
| 1) The actual parameter is never modified by execution of the called function. | 5) all of the above  |
| 2) The formal parameter is never modified by execution of the called function. | 6) 2 and 3 only      |
| 3) The actual parameter must be a variable.                                    | 7) 1 and 2 only      |
| 4) The actual parameter cannot be of type <code>bool</code> .                  | 8) 1 and 3 only      |
|  | 9) none of the above |
- 

For the next three questions, consider execution of the following program:

```
void DoThis(int& Alpha, int Beta);  
void main() {  
    int Temp = 15;  
    int Ben = -5, Jerry = 42;  
    DoThis(Ben, Jerry);  
    cout << "Ben   = " << Ben << endl;  
    cout << "Jerry = " << Jerry  
        << endl;  
    cout << "Temp  = " << Temp << endl;  
}  
  
void DoThis(int& Alpha, int Beta) {  
    int Temp;  
    Alpha = Alpha + 100;  
    Temp = Beta;  
    Beta = 999;  
    Temp = 12;  
}
```

17) What value is printed for the variable `Ben`?

- |        |                      |       |       |        |
|--------|----------------------|-------|-------|--------|
| 1) -5  | 2) 12                | 3) 42 | 4) 95 | 5) 142 |
| 6) 999 | 8) None of the above |       |       |        |

18) What value is printed for the variable `Jerry`?

- |        |                      |       |       |        |
|--------|----------------------|-------|-------|--------|
| 1) -5  | 2) 12                | 3) 42 | 4) 95 | 5) 142 |
| 6) 999 | 8) None of the above |       |       |        |

19) What value is printed for the variable `Temp`?

- |        |                      |       |       |        |
|--------|----------------------|-------|-------|--------|
| 1) -5  | 2) 12                | 3) 42 | 4) 95 | 5) 142 |
| 6) 999 | 8) None of the above |       |       |        |

20) If a variable Alpha is accessible only within function F, then Alpha is either:

- 1) a global variable or a formal parameter of F
  - 2) a local variable within F or a formal parameter of F
  - 3) a global variable or an actual parameter to F
  - 4) a local variable within F or an actual parameter to F
  - 5) none of the above
- 

21) What is the most appropriate function prototype for a function that receives a character letter grade and returns its integer equivalent on a four-point grading scale?

- |                           |                          |
|---------------------------|--------------------------|
| 1) void IntEquiv(char );  | 6) int IntEquiv(char& ); |
| 2) void IntEquiv(char& ); | 7) all of the above      |
| 3) void IntEquiv(int );   | 8) 1 and 5 only          |
| 4) void IntEquiv(int& );  | 9) 5 and 6 only          |
| 5) int IntEquiv(char );   | 10) none of the above    |
- 

22) What is the output of the following code fragment?

```
int Limit = 10;
cout << 'H';
for (int LoopCount = 8; LoopCount < Limit; LoopCount++)
    cout << 'E';
cout << "LP";
```

- |          |                      |
|----------|----------------------|
| 1) HLP   | 4) HEEELP            |
| 2) HELP  | 5) none of the above |
| 3) HEELP |                      |
- 

23) What loop condition should replace “? ? ?” in the code below in order to produce the output: 8 17 35 71?

```
int N = 8;
do {
    cout << setw(3) << N;
    N = N*2 + 1;
} while ( ? ? ? );
```

- |            |                      |
|------------|----------------------|
| 1) N <= 71 | 4) N >= 8            |
| 2) N < 71  | 5) N > 143           |
| 3) N < 35  | 6) none of the above |
- 

24) Given the array declaration below, what is the range of valid index values for myArray[ ]?

```
char myArray[75];
```

- |                 |                      |
|-----------------|----------------------|
| 1) 0 through 75 | 5) 1 through 74      |
| 2) 0 through 74 | 6) 1 through 76      |
| 3) 0 through 76 | 7) none of the above |
| 4) 1 through 75 |                      |
-

25) Given the declarations: `int Status[10];`  
which of the following loops correctly initializes the array to hold all zeros?

- |   |                      |
|---|----------------------|
| 1) <code>for (int i = 0; i &lt; 10; i++)<br/>    Status[i] = i;</code>  | 5) all of the above  |
| 2) <code>for (int i = 0; i &lt; 10; i++)<br/>    Status[i] = 0</code>   | 6) 1 and 2 only      |
| 3) <code>for (int i = 0; i &lt;= 10; i++)<br/>    Status[i] = 0</code>  | 7) 2 and 3 only      |
| 4) <code>for (int i = 0; i &lt;= 10; i++)<br/>    Status[i] = i;</code> | 8) none of the above |
- 

26) What is the output of the following program?

```
#include <iostream.h>
void main() {
    int Ray[5] = {100,200,300,400,500};
    int k;
    for (k = 4; k >= 0; k--)
        cout << Ray[k] << ' ';
}
```

- |                        |                      |
|------------------------|----------------------|
| 1) 100 200 300 400 500 | 5) 5 4 3 2 1         |
| 2) 400 300 200 100 0   | 6) 400 300 200 100   |
| 3) 500 400 300 200 100 | 7) 100 200 300 400   |
| 4) 4 3 2 1 0           | 8) none of the above |
- 

27) Given an integer array `Fred[]` holding 5000 values, which of the code fragments below could be used to print out the values of `Fred[0]`, `Fred[2]`, `Fred[4]`, etc:

- |  |                       |
|--|-----------------------|
| 1) <code>for (int i = 0; i &lt; 5000; i = i + 2)<br/>    cout &lt;&lt; Fred[i] &lt;&lt; endl;</code> | 6) all of the above   |
| 2) <code>for (int i = 0; i &lt; 2500; i++)<br/>    cout &lt;&lt; Fred[i] &lt;&lt; endl;</code>       | 7) 1 and 2 only       |
| 3) <code>for (int i = 0; i &lt; 2500; i++)<br/>    cout &lt;&lt; Fred[2*i] &lt;&lt; endl;</code>     | 8) 1 and 3 only       |
| 4) <code>for (int i = 0; i &lt; 2500; i++)<br/>    cout &lt;&lt; 2*Fred[i] &lt;&lt; endl;</code>     | 9) 1, 2, and 3 only   |
| 5) <code>for (int i = 0; i &lt; 5000; i++)<br/>    cout &lt;&lt; Fred[i] &lt;&lt; endl;</code>       | 10) none of the above |
-

28) Given the declaration: `char myName[4] = "Ben";`

which of the following would not output "Ben" (without the quotes, of course, and no extra characters)?

- |   |                            |
|---|----------------------------|
| 1) <code>cout &lt;&lt; myName;</code>   | 5) none of the above would |
| 2) <code>for (int ch = 0; ch &lt; 3; ch++)<br/>    cout &lt;&lt; myName[ch];</code> | 6) 1 and 2 would not       |
| 3) <code>for (int i = 0; i &lt; 3; i++)<br/>    cout &lt;&lt; myName;</code>        | 7) 3 and 4 would not       |
| 4) <code>for (int i = 0; i &lt; 4; i++)<br/>    cout &lt;&lt; myName[i];</code>     | 8) all of the above would  |

---

For the next four questions, consider writing a program that contains the following variable declarations and function prototypes.

```
const int MaxSize = 5;
void FillWholeArray(int Ray[], int arrayDimension);
int  aRay[MaxSize] = {1, 2, 3, 4, 5},
     bRay[MaxSize] = {27, 13, 4, 4, 2},
     RayStevens[MaxSize-1] = {0, 0, 0, 0};
```

Indicate whether the proposed statement given in each question is:

- 1) syntactically illegal; i.e., there would be a compile-time error
- 2) syntactically legal, but logically incorrect
- 3) syntactically legal and logically correct, as far as we can tell

29) `aRay[MaxSize] = bRay[MaxSize];`

30) `FillWholeArray(aRay[MaxSize], MaxSize);`

31) `FillWholeArray(aRay, 4);`

32) `aRay[3] = RayStevens[bRay[4]];`

---



For the next five questions, consider the incomplete function definition given below:

```
// MaxValue() takes an array of integers and returns the index of the
// largest value in the array.
//
// Parameters:
//     List[]    array of integers
//     HowMany   number of values stored in List[]
// Returns:
//     the index of the largest value in List[]
//
int MaxValue(_____ List[], _____ HowMany) { // line A

    int maxSoFar = _____;           // line B
    int Look;                                     // line C
    for (Look = 1; _____; Look++)      // line D
        if (List[maxSoFar] < List[Look])    // line E
            _____;                   // line F

    return maxSoFar;                           // line G
}
```

33) How should the blank preceding the first parameter in line A be filled?

- 1) int                      2) const int                      3) int&
- 4) const int&              5) none of the above

34) How should the blank preceding the second parameter in line A be filled?

- 1) int                      2) int&                      3) const int&
- 4) none of the above

35) How should the blank in line B be filled? Be careful.

- 1) 0                      2) -1                      3) List[0]
- 4) HowMany              5) none of the above

36) How should the blank in line D be filled?

- 1) Look <= HowMany              2) Look < HowMany              3) Look < maxSoFar
- 4) Look == HowMany              5) none of the above

37) How should the blank in line F be filled?

- 1) Look = maxSoFar              2) Look == maxSoFar              3) maxSoFar = Look
  - 4) maxSoFar = List[Look]      5) none of the above
-

For the following two questions, assume the following declarations:

```
enum BigEast { BC, Miami, Pitt, Rutgers, Syracuse, Temple, VaTech, WVU };  
  
BigEast Num1Team = VaTech;
```

38) Which of the following outputs the team that is stored in Num1Team by the initialization?

- 1)    if (Num1Team == "VaTech")  
      cout << Num1Team;
- 2)    if (Num1Team == VaTech)  
      cout << "Num1Team";
- 3)    if (Num1Team == "VaTECH")  
      cout << "VaTech";
- 4)    if (Num1Team == VaTech)  
      cout << "VaTech";
- 5)    None of the above

39) What value is output by the following code:

```
cout << int (Num1Team) << flush << endl ;
```

- 1) 1    2) 2    3) 3    4) 4    5) 5
- 6) 6    7) 7    8) 8    9) 0    10) None of the above

---

40) What value is output by the following code:

```
cout << strcmp ( "WVU" , "VaTech" );
```

- 1) a positive integer                   2) 0                   3) a negative integer
  - 4) "VaTech"                           5) "WVU"               6) "WVUVaTech"
  - 7) None of the above
-