1) **Which of the following is the "brain" of a computer?**

(a) RAM
(b) GPU
(c) CPU
(d) SSD

2) **What does the acronym “CPU” mean?**

(a) Committee of Programming Users
(b) Course Provided Utilities
(c) Command Processing Unit
(d) Central Processing Unit

3) **Which of the following statements is FALSE about "machine code"?**

(a) machine code is stored as binary (zeros and ones)
(b) C++ is an example of it
(c) machine code is typically generated by a compiler
(d) all programs executed on a computer are really machine code

4) **What is a benefit of a high level language over machine code?**

(a) A high level language runs in high memory, so it is faster
(b) A high level language allows the programmer to describe tasks using a syntax that is more closely related to the problems to be solved
(c) Similar to a comparison of High German and Low German, a high level language is “spoken” by more people
(d) A high level language is dependent on the processor

5) **Programming includes all of the following EXCEPT _____.**

(a) understanding the problem to be solved
(b) determining how to solve the problem
(c) writing code to solve the problem
(d) (no exceptions here)

6) **What us the function of a compiler?**

(a) to provide an integrated environment where programs can be created, edited, debugged, run, tested, etc.
(b) to produce machine code from source code written in C++, for example
(c) to build an executable (i.e. runnable) file
(d) to identify errors in source code

7) **How can you a achieve the efficiency of a reference parameter and the meaning of a value parameter?**

(a) make a parameter a constant reference parameter
(b) pass (into the function) a dummy variable whose value is set by the function
(c) use the return value for passing the value into and out of the function
(d) you can't; it's one or the other, but never both

8) **Pick the INCORRECT statement:**

(a) The private data members of a class are accessible and modifiable by the functions of the same class.
(b) One function of a class can call another function of the same class.
(c) Functions included in the public part of a class may be called directly using objects of the class.
(d) Functions included in the private part of a class may be called directly using objects of the class.

9) **Which of the following is *NOT* a valid identifier:**

(a) 7daysinaweek
(b) days_in_a_week
(c) daysInAWeek
(d) DAYS_IN_A_WEEK

10) **Which of the following is not a good variable name?**

(a) length
(b) firstName
(c) x
(d) score

11) **Which of the following is NOT a valid type keyword?**

(a) double
(b) integer
(c) float
(d) char

12) **Which of the following lines contains a syntax error:**

(a) `cout << "Do you want" << " to build a snowman?";`
(b) `cout << "Do you want to build a snowman?" << endl`
(c) `cout << "Do " << "you" << " want to build a " << "snowman?";`
(d) `cout << "Do you want to build a snowman?";`

13) **Which of the following would *NOT* typically be found in pseudocode?**

(a) repeat
(b) while
(c) for each
(d) namespace

14) **How many backslashes (\) are displayed as a consequence of**
`cout << "\\\" Hello,\" backslashes!" << endl;`

(a) none
(b) 1
(c) 2
(d) 4
(e) (this will not compile)

15) **What is wrong with the following line of code?**

```
cout << celcius_temp * 9/5 + 38 << endl
```

(a) It is syntactically incorrect.
(b) It is logically incorrect.
(c) It is both logically and syntactically incorrect.

16) **Which of the following variable names is unacceptable?**

(a) CS142_quiz
(b) CS_142_quiz_question
(c) 142_quiz
(d) double_quiz

17) **The following code snippet contains _____ error(s).**

```
double my_score;
const double BONUS = 0.10;
my_score = my_score * BONUS;
cout << my_score << endl;
```

(a) no
(b) one or more syntax
(c) one or more logic
(d) one or more syntax and/or logic

18) **What is displayed by**

```
cout << 365/52 << " days in a week";
```

(a) 7days in a week
(b) 7 days in a week
(c) 7.019230days in a week
(d) 7.019230 days in a week

19) **Which of the following is needed for**

```
abs(sqrt(pow(a,2) + pow(2,b));
```

(a) #include << cmath >>
(b) #include << cstdlib >>
(c) #include << cmath >>
    #include << cstdlib >>
(d) #include << cmath >>;
    #include << cstdlib >>;

20) **Which following is the best description of the manipulator setprecision()?**

(a) changes a variable from float to double
(b) changes the width of an output
(c) changes how many digits after the decimal are output
(d) changes how many digits are output

21) **Which of the following is a correct order of operator precedence (highest on the left)?**

(a) + - () * /
(b) () * / + -
(c) * / + - ()
(d) * / () + -

22) **Which of the following is NOT a valid usage of an if statement? (Remember that using multiple lines is good practice but not required by the compiler)**

(a) `x = y-10; if (x > 0) { positive = true; }`
(b) `age++; if (age < 16) { can_drive = false; } else { can_drive = true; }`
(c) `if (wheels > 3) type = "car";`
(d) `cin >> wheels; type = "bike" if (wheels < 3);`

23) **Which of the following lines could you use to join two strings together?**

(a) `substr()`
(b) `length()`
(c) `merge()`
(d) `+`
(e) `attach()`

24) **To obtain input from the user via the keyboard, use**

(a) `cout >> ...;`
(b) `cin << ...;`
(c) `cout << ...; cin >>...;`
(d) `cout >> ...; cin << ...;`

25) **Which of the following affects only the next value?**

(a) `cout << fixed`
(b) `cout << setprecision(3)`
(c) `cout << setw(7)`
(d) all affect only the next value

26) **Which is the following is disallowed in C++?**

(a) `string couple = "Pinta" + " " + "Judges";`
(b) `string couple = couple + "Ruth";`
(c)
```
string his, hers, juniors;
cin >> his >> hers >> juniors;
```
(d) All are allowed

27) **What is displayed by**

```
string sentence = "Tomorrow!";
cout << sentence.substr(sentence.length()/2,3);
```

(a) error
(b) rro
(c) row
(d) ow!

28) **Which of the following can be critical in input validation?**

(a) `cout << "try again";`
(b) `cin.fail()`
(c) `the = operator`

29) **Which of the following comparison operators do you need to be extra careful with?**

(a) <
(b) ==
(c) >
(d) !=
(e) <=

30) **Which of the following is NOT a valid while loop?**

(a) `int x = 10; while (x > 0) cout << "counting down..."; x--;`
(b)
```
int x = 2; while (x <= 10)
  { cout << "Counting evens, " << x << endl; x+= 2; }
```
(c) `bool valid = false; while (!valid) { cout << "."; }`

31) **Which of the following types *CANNOT* be passed as the decision variable** (switch (*variable*) { case ... }) **to a switch statement?**

(a) string
(b) char
(c) int

32) **Which of the following words is the name for the way that strings are ordered, i.e. the way that strings are compared using >, <, etc.?**

(a) alphabetic
(b) numeric
(c) lexicographic
(d) alphanumeric

33) **In the following code snippet**

```
if (floor > 13)
{
     actual_floor = floor – 1;
}
```

the braces are:

(a) unnecessary
(b) unnecessary, but a good idea
(c) necessary
(d) problematic and in conflict with the braces for the body of the main function

34) **What is displayed by the following code snippet?**

```
int floor = 9;
if (floor >= 13);
{
     floor--;
}
cout << floor;
```

(a) 8
(b) 9
(c) it will not compile
(d) it will produce a run-time error

35) **Which of the following is first lexicographically?**

(a) 999
(b) Zeezrom
(c) awful
(d) 7-11

36) **What is the difference between A and B**

A:
```
if (condition1)
     statement1
else
     if (condition2)
          statement2
```
B:
```
if (condition1)
     statement1
else if (condition2)
          statement2
```
(a) statement2 is executed by B only if condition1 is false
(b) statement2 is executed by B only if condition2 is true
(c) statement3 is executed by B only if condition1 is false
and condition2 is true
(d) there is no functional difference between A and B

37) **What will be displayed by**

```
int temp = 12;
if (65 <= temp <= 80)
   cout << "Nice day!";
else
   cout << "Not-so-nice day!";
```

(a) Nice day!
(b) Not-so-nice day!
(c) (nothing will be displayed)
(d) (it will not compile)

38) **What will be displayed by**

```
bool give_me_liberty = true;
bool give_me_death = true;
bool escape = give_me_liberty || give_me_death;
if (escape) cout << "Patrick will be home\n";
else cout << "Patrick can't make it\n";
```

(a) Patrick will be home
(b) Patrick can't make it
(c) Patrick will be home
Patrick can't make it
(d) (this will not compile)

39) **Which of the following for loops will iterate exactly 10 times?**

(a) `for (int i = 1; i < 10; i++){...}`
(b) `for (int i = 0; i < 10; i++){...}`
(c) `for (int i = 0; i <= 10; i++){...}`
(d) `for (int i = 1; i > 10; i++){...}`
(e) `for (int i = 1; i == 10; i++){...}`
(f) None of the above

40) **What would the following code snippet output?**

```
int start = 5;
while ( start > 0 ) {
    if ( start%2==1 ) { cout << start; }
    start--;
}
```

(a) 54321
(b) 531
(c) 420
(d) 42
(e) 543210

41) **Which of the following loops will always execute at least once?**

(a) do while
(b) for
(c) while
(d) None of the above

42) **Which loop would you want to use if you are writing a program that will do something repeatedly and you know beforehand exactly how many times the program will need to do it?**

(a) for
(b) while
(c) do-while

43) **Which of the following would generate a random integer between 1 and 6, inclusive?**

(a) `int dieRoll = ( rand() % 7 );`
(b) `int dieRoll = ( rand() % 6 );`
(c) `int dieRoll = ( rand() % 7 -1 );`
(d) `int dieRoll = ( rand() % 6 +1 );`

44) **Which of the following is equivalent to**
`(!(A && (B || C))`

(a) `!A && (!B || !C)`
(b) `!A || (!B || !C)`
(c) `!A || !(B || C)`
(d) `!A || !(B && C)`

45) **If your program needs a floor between 0 and 18, but not equal to 13, what should you verify second?**

(a) that the floor is not 13
(b) that the floor is between 0 and 18
(c) either (a) or (b)
(d) that the floor is an int

46) **Which is the right question for determining if you should continue taking classes?**

(a) while (credit_hours > 120)
(b) while (credit_hours < 120)
(c) while (credit_hours >= 120)
(d) while (credit_hours <= 120)

47) **What is displayed by**

```
int i = 5;
while (i >= 0);
{
      cout << i << " ";
      i--
}
```

(a) 5 4 3 2 1
(b) 5 4 3 2 1 0
(c) 4 3 2 1 0
(d) nothing

48) **How many times will the following loop iterate?**

```
for (int i = -836; i <= 743; i ++) { ... }
```

(a) 836 - 743
(b) 743 + 836
(c) 743 - (-836) + 1
(d) none

49) **You've received 6 job offers already and you expect to receive more offers. However, you can evaluate only the first 4 offers you received. You should use _____ loop.**

(a) a do
(b) a for
(c) a while
(d) either a do or a while

50) **Given the definition below of the function sum:**

```
int sum(int a, int b)
{
    return (a+b);
}
```

Which one of the following would result in the number 7 being printed?

(a) `cout << sum(7,1);`
(b) `cout << sum(5+2);`
(c) `cout << sum(1,6);`
(d) `cout << sum(1,2,4);`

51) **In the function below, what do we call the part in italics?**

```
int something(int a)
{
    // one or more statements...
}
```

(a) function name
(b) return type
(c) function body
(d) parameter

52) **Which of the options below completes the function sumRange? This function should total all the integers between lowest and highest inclusive, for example sumRange(5,8) would be the sum of 5, 6, 7, and 8.**

```
int sumRange(int lowest, int highest)
{
    // INSERT MISSING LINE(S) HERE
    {
        sum += current;
    }
    return sum;
}
```

(a)
```
int sum;
for( int current = 0; current <= highest; current++ )
```

(b)
```
int sum;
for( int current = 0; current < highest; current+=lowest )
```

(c)
```
int sum = lowest;
for( int current = 0; current < highest; current++ )
```

(d)
```
int sum = 0;
for( int current = lowest; current <= highest; current++ )
```

53) **Which of these is NOT a valid function**
(not whether it will give an intended result, but whether it will compile)?

(a)
```
int max(int a, int b)
      { if (a>b) {return a;} else {return b;} }
```

(b)
```
double average(int a, int b)
      { double avg = (a+b)/2.0; }
```

(c)
```
char getChar(string s, int i)
      { return s.substring(i,1); }
```

(d)
```
string longer(string s, string t)
      { if (s.length() > t.length()) { return s;} else { return t;} }
```

54) **How do you escape from the following loop?**

```
string response;
string input;
while (cin >> input)
     response += input;
cout << response;
```

(a) Enter a negative value for input
(b) Enter a value which is not a string
(c) Make the condition of the while loop a compound condition
(d) You can’t escape

55) **Assume the following snippet displays 4 3 3 5 1 6 when we run it today.**
Which of the following will it most likely display when we run it tomorrow?

```
int main()
{
      for (int i = 1; i <= 6; i++)
      {
                int r = rand() % 6 + 1;
                cout << r << “ “;
      }
      return 0;
}
```

(a) 3 3 0 5 4 1
(b) 4 7 1 3 1 2
(c) 6 3 -2 1 4 5
(d) 4 3 3 5 1 6

56) **Suppose that an accrediting agency requires us to write a program to find the average score of all students in all sections of CS 142 for the past 6 semesters. This program would use _____.**

(a) a loop
(b) a loop inside a loop
(c) a loop inside a loop inside a loop
(d) a loop inside a loop inside a loop inside a loop

57) **Which of the following will generate a random number between 0 and 31415?**

(a) rand() * 31415
(b) rand() * 31414 +1
(c) rand() % 31415
(d) rand() % 31416

58) **From the users point of view, the most important thing(s) to know about a function is/are**

(a) what it needs
(b) what it returns
(c) what it needs and what it returns
(d) what algorithm it uses

59) **If a program consists of one or more functions (in addition to the main function), _____.**

(a) the main function should be at the top
(b) the main function should be at the bottom
(c) the functions should be ordered alphabetically (by function name)
(d) the order is of no consequence

60) **Pick the best statement.**

(a) all functions have 1 or more formal parameters
and a single return value.

(b) all functions have 0 or more formal parameters
and a single return value.

(c) all functions have 1 or more formal parameters
and 1 or more return values.

(d) all functions have 0 or more formal parameters
and some functions have no return value

61) **When should you declare your own global variables?**
(a) when the same variable is needed by more than one function
(b) when a variable is declared with public visibility
(c) when the number of variables needed
by a function would otherwise exceed 3
(d) probably never

62) **A function typically should be broken up into smaller functions when ____.**

(a) it won't fit on a single screen
(b) the task it is to perform can be broken up
into smaller, nontrivial tasks
(c) one or more of the tasks it performs also needs
to be performed by one or more other functions
(d) (all of the above)

63) **What is the purpose of a stub?**

(a) to facilitate the calling of a function
which has not yet been developed - typically for testing purposes
(b) to provide a function with an empty function body
(c) to restrict the range of parameters that can be passed to a function
(d) to provide a function which does not return a value

64) **I'm interested to learn about versification in your "First Edition" of The Book of Mormon. You can choose to (a) find another "First Edition" and give it to me or (b) tell me the location of of your "First Edition" so I can access in. In (a) you are passing the Book to me by _____; in (b) you are making it available to me by _____.**

(a) reference, reference
(b) reference, value
(c) value, reference
(d) value, value

65) **Assuming we have this function,**

```
int squared( int x )
{
    return x*x;
}
```

What will be printed by the code below?

```
int result;
for (int i = 2; i < 5; i++)
{
    result += squared (i);
}
cout << result;
```

(a) 29
(b) 54
(c) 4 9 16
(d) 4916
(e) 4 9 16 25

66) **Carefully select the proper way to call the function below and store the result:**

```
double averageOf(int x, int y, int z)
{
    return (x + y + z)/3.0;
}
```

(a) double avg = averageOf(1+2, 5, 2);
(b) int avg = averageOf(2, 4, 5);
(c) double avg = averageOf(10, 20, 30, 40);
(d) double avg = averageOf(98, 99);

67) **Given the array declaration:**

```
int values[] = { 45, 19, 23, 17, 12, 24, 99, 16 };
```

which of the following represents the smallest element in the array?

(a) values[0]
(b) values[4]
(c) values[5]
(d) values.min()

68) **Given the following statements about arrays:**

1) arrays can be resized
2) arrays can be initialized when declared
3) a single array can hold a mixture of types

which answer below uniquely indicates
the set of statements which are true?

(a) 1 only
(b) 2 only
(c) 1 and 2
(d) 2 and 3
(e) 2 only

69) **Suppose you're writing a function myfunction()...**
which of the following would NOT be a reason
to use a pass-by-reference parameter?

(a) if myfunction() needs to return
multiple pieces of information

(b) when another function will refer to the
value that is returned by myfunction()

(c) myfunction() needs to change the value
of a variable in the function that calls it

70) **What type of value will the following function return?**

```
//    timesTwo takes a number and multiplies
//    it by two, and returns the result

int timesTwo(int inNum){...}
```

(a) bool
(b) char
(c) double
(d) float
(e) int
(f) string
(g) void
(h) Something not listed above
(i) It does not have a return value

71) **Which of the following code snippets would satisfy the description of the function in the previous question?**

(a) int rtn = inNum + 2; return rtn;
(b) int rtn *= inNum * 2; return rtn;
(c) return inNum * 2;
(d) return inNum + 2;
(e) All of the above
(f) None of the above

72) **The advantage of passing a reference parameter as a const reference is**

(a) to protect the parameter from being changed by the function

(b) to notify the user that the parameter cannot be changed by the function

(c) to save having to make a copy of the parameter

(d) all of the above

73) **What is displayed if the value 5 is passed into**

```
void chip_away(int n)
{
   if (n < 1) {return;}
   chip_away(n - 1);
   cout << n << " ";
}
```

(a) 1 2 3 4 5
(b) 5 4 3 2 1
(c) 0 1 2 3 4
(d) 4 3 2 1 0

74) **You can have an array of all the following EXCEPT**

(a) ints
(b) strings
(c) ints and strings together
(d) (no exceptions here)

75) **Which of the following defines an array consisting of five integers?**

(a) int myArray[] = {0, 1, 2, 3, 4};
(b) int myArray[5];
(c) const int ARRAY_SIZE = 5; int myArray[ARRAY_SIZE];
(d) int arraySize = 5; int myArray[arraySize];
(e) a, b, and c
(f) a, b, and d
(g) a, c, and d
(h) b, c, and d
(I) a, b, c, and d

76) **What is/are the consequence(s) of executing the following code snippet:**

```
int my_scores[10];
my_scores[10] = 100;
cout << my_scores[10];
```

(a) memory is corrupted
(b) 100 is displayed
(c) memory is corrupted and 100 is displayed
(d) the code will not execute

77) **What is the best way to assign to array my_stuff the contents of array your_stuff?**

(a) my_stuff = your_stuff;
(b) my_stuff[] = your_stuff[];
(c) with a for loop containing my_stuff[i] = your_stuff[i];
(d) with a while loop containing my_stuff[i] = your_stuff[i];

78) **To remove an element at position pos in an ordered array her_stuff, _____, and decrease the size of the array by one.**

(a) copy the element from the end of array her_stuff
into the element at position pos

(b) move everything one element closer to the
beginning of array her_stuff, starting with the element
at position pos + 1

(c) move everything one element closer to the beginning of
array her_stuff, starting with the element
at the end of array her_stuff

(d) (you can't do it; once her_stuff is created, it cannot be modified)

79) **Which of the following can a vector do but an array cannot?**

(a) tell you how many elements are in it
(b) store a new element
(c) proved a requested element
(d) add all of its elements together for you

80) **Which of the following is used for writing data to a file?**

(a) a savefilestream
(b) an iostream
(c) a writestream
(d) an ofstream

81) **What is the relationship between the capacity of an array and the current size of that array?**

(a) capacity and current size are synonyms
(b) capacity is how much it can hold; current size is how much it does hold
(c) current size is how much it can hold; capacity is how much it does hold
(d) capacity refers to the size of the individual elements
(doubles or or ints or ...);
current size refers to the maximum number of elements
that can be placed in the array

82) **Pick the best statement:**
(a) the return type of a function can be an array
(b) an array cannot be passed into a function, but an element of an array can be passed into a function
(c) an array can be passed into (and out of) a function by reference
(d) an array is passed into a function by reference, using the &

83) **Which of the following function headers is best for totaling all the elements in an array of int?**

(a) int table_total(int table[][], int rows, int columns)
(b) int table_total(int table[ROWS][], int columns)
(c) int table_total(int table[][COLUMNS], int rows)
(d) int table_total(int table[ROWS][COLUMNS])

84) **cout << values.size() suggests that values is**

(a) an array
(b) a vector
(c) either an array or a vector
(d) neither an array nor a vector

85) **Vectors cannot be _____.**

(a) passed into a function by value
(b) returned by a function
(c) passed into a function without an accompanying size parameter
(d) copied with a single line of code such as my_vector = your_vector;
(e) everything here can be done with vectors

86) **Pick the INCORRECT statement:**

(a) Files can be opened for reading only if they exist.
(b) Files can be opened for writing if they exist.
(c) Files can be opened for writing if they do not exist.
(d) Files can be opened for reading if they do not exist.
(e) None of these statements is incorrect.

87) **Which of the following is the difference between arrays and vectors that one needs to be most concerned with when passing them as parameters to functions?**

(a) vectors "know" their own size, arrays don't.
(b) arrays have a fixed size, vectors are resizable.
(c) arrays don't need an "&" to pass by reference
(d) arrays can be created with empty elements

88) **If in_file contains**

```
10 Joseph 260365 1.2681
```

what is displayed by

```
string next_string;
in_file >> next_string;
char next_ch;
in_file.get(next_ch);
in_file.get(next_ch);
cout << next_ch;
```

(a) 1
(b) (space)
(c) J
(d) o

89) **Assuming in_file is the same as in the previous question, what is displayed by**

```
string line;
getline(in_file, line);
cout << line.substr(3,4);
```

(a) (space)J
(b) (space)Jos
(c) Jo
(d) Jose

90) **Assuming out_file is an ofstream, all of the following are acceptable EXCEPT**

(a) out_file.put('a');
(b) out_file << "fixed";
(c) out_file << endl;
(d) (all are acceptable)

91) **Once you have defined a class, you can _____.**

(a) call the functions of that class.
(b) create just one object of that class.
(c) create as many objects of that class as you want.
(d) associate accessors (getters), mutators (setters), and
data members with the class

92) **Why separate the interface (the "public" part of a class) from the implementation of the functions listed in the interface?**

(a) To separate "what" can be done from "how" it is done
(b) To protect the user from detail that he neither needs nor wants
(c) To make it possible to change the implementation (the "how")
without changing the interface (the "what")
(d) All of the above

93) **Why have classes and objects?**

(a) So you can have variables of types not typically included in C++
(b) To bring together the data and the functions that work on the data
(c) To manage large collections of functions and data
(d) All of the above

94) **Given this code:**

```
int a[] = {1,2,3};
int* b = a;
```

The following two statements are equivalent...

```
cout << a[1];
cout << *(b+1);
```

TRUE or FALSE?

(a) TRUE
(b) FALSE

95) **Which of the following exemplifies accessing the value held by a pointer?**

(a) `int* x = NULL;`
(b) `int* a = &e;`
(c) `int b = *d;`
(d) `int k = 0x00fd37ce;`

96) **Which of the following is the best reason to use inheritance in C++?**

(a) To provide common functionality shared by two or more classes
(b) To make code run faster
(c) To use someone else's code
(d) To write family history software

97) **Which of the following will remove the 4th item in a Vector of 7 items (where the vector is held in a variable called vec)?**

a) `vec.remove(4);`
b) `vec.erase(3);`
c) `vec.erase( vec.begin()+3);`
d) `vec.remove( 4*vec.begin() );`

98) **Pick the INCORECT statement:**

(a) The purpose of a constructor is to initialize the data members.
(b) After an object is created, the constructor can be called to re-initialize that object's data members.
(c) The constructor has no return type, not even void.
(d) You can have multiple constructors, all with the same name.

99) **Pick the INCORRECT statement(s):**

(a) Functions of a class can access and modify the private data members of the class.

(b) Functions of a class can call other functions of the same class, whether public or private.

(c) An object of a class can be used to call the functions of that class directly, even if the functions are private.

(d) All the functions of a class must be public.

100) **Given**

```
double my_gpa = 3.95;
double* gpa_pointer = &my_gpa;
```

which of the following is LEAST meaningful?

(a) `cout << my_gpa;`
(b) `cout << gpa_pointer;`
(c) `cout << *gpa_pointer;`
(d) `cout << *(&my_gpa);`

101) **Continuing the previous question, which of the following is LEAST meaningful?**

(a) `my_gpa += 0.03;`
(b) `gpa_pointer += 0.03;`
(c) `*gpa_pointer += 0.03;`
(d) `*(&my_gpa) += 0.03;`

102) **How many pointers to ints are created by:**

```
int* p1, p2, p3, p4;
```

(a) None
(b) One
(c) Four
(d) This line of code will not compile

103) **Assume that array my_array contains the first 5 prime numbers, stored as ints. What is displayed by**

```
cout << *my_array + 3;
```

(a) 2
(b) 3
(c) 5
(d) 7

104) **Which class does the following function belong to?**
`double Donut::get_weight() const{ return weight; }`

(a) double
(b) Donut
(c) get_weight()
(d) const
(e) It cannot be determined with the given information

105) **What would the output of this code be?**

```
int x[] = {7,8};
int* y = x;
y++;
*y += 4;
cout << x[0] << " " << x[1];
```

(a) 7 8
(b) 11 8
(c) 11 12
(d) 7 12

106) **Given two classes, Employee and Manager, which would be the base class?**

(a) Employee
(b) Manager

107) **Which of the following would be a derived class for Dog?**

(a) Canine
(b) Cat
(c) Poodle
(d) Fluffy

Questions 108 – 112 refer to the following the class and variable definitions:

```
class ClassA
{
public:
    string toString() { return "A"; }
};

class ClassB: public ClassA
{
public:
    string toString() { return "B"; }
};

class ClassC
{
public:
    virtual string toString() { return "C"; }
};

class ClassD: public ClassC
{
public:
    string toString() { return "D"; }
};

ClassB* bptr = new ClassB();
ClassA* aptr = bptr;
ClassD* dptr = new ClassD();
ClassC* cptr = dptr;
```

108) **What would this print?**

```
cout << aptr->toString();
```

(a) A
(b) B
(c) C
(d) D

109) **What would this print?**

```
cout << bptr->toString();
```

(a) A
(b) B
(c) C
(d) D

110) **What would this print?**

```
cout << cptr->toString();
```

(a) A
(b) B
(c) C
(d) D

111) **What would this print?**

```
cout << dptr->toString();
```

(a) A
(b) B
(c) C
(d) D

112) **Which of the following describes the relationship between a base class B and a derived class D?**

(a) B is a D
(b) D is a B
(c) B has a D
(d) D has a B

113) **Which is the proper way to access the fifth element of my_array (which contains ints, each of which takes 4 bytes)?**

(a) my_array + 5
(b) my_array + 5*4
(c) *my_array + 5
(d) *(my_array + 5)

114) **If a byte is sufficient to hold a single character, how many bytes are needed for the array created by**

```
const char* course_name = "CS 142";
```

(a) 5
(b) 6
(c) 7
(d) None; The assignment is not valid

115) **Source files might include a .h file and a .cpp file for each class, and a .cpp file containing the main function. All of the following are reasons to separate these source files EXCEPT**

(a) eliminating the need to recompile files that have not been changed
(b) partitioning the solution into parts that can be worked on separately
(c) elimination of some of the code that is needed when all these files are together in a single source file.
(d) no exceptions here

116) **Suppose your program contains**

```
string* my_strings = new string[];
```

Later on in your program, there should be _____.

(a) delete my_strings;
(b) delete[] my_strings;
(c) delete my_strings*;
(d) delete my_strings[]*;

117) **Pick the unnecessary or inappropriate practice or statement:**

(a) Use the word "new" to explicitly allocate new values at run time.
(b) Every use of "new" should be accompanied by a subsequent use of "delete."
(c) You often can, but should never access a memory block after it has been deleted.
(d) All these statements are appropriate or necessary.

118) **Which of the following lines is improper?**

(a) `double my_num = 3.5;`
(b) `double* my_pointer = &my_num;`
(c) `*my_pointer = 3.7;`
(d) `delete my_pointer;`

119) **gpa_pointer->increase_gpa(3.85) is equivalent to**

(a) gpa_pointer.increase_gpa(3.85)
(b) this.increase_gpa(3.85)
(c) *gpa_pointer.increase_gpa(3.85)
(d) *(gpa_pointer).increase_gpa(3.85)

120) **Class BankAccount has three data members (name, account_number, and balance) and two member functions (deposit and withdraw).**

Class SavingsAccount extends class BankAccount and declares data member interest_rate and member function post_interest.

How many data members and member functions, respectively, does an object of class SavingsAccount have?

(a) 1 and 1
(b) 1 and 3
(c) 4 and 1
(d) 4 and 3

121) **Given**

```
class ChildClass : public ParentClass
```

what is the proper was to pass a parameter to ParentClass's constructor?

(a) ChildClass::ChildClass(string param) { ParentClass(param);}
(b) ChildClass::ChildClass(string param) : ParentClass(param)
(c) ChildClass::ChildClass(string param) {ParentClass::ParentClass(param);}
(d) You can't pass a parameter to ParentClass's constructor

122) **Assume ChildClass and ParentClass are related as in the previous question. Assume further than each class has a function DoIt which takes no parameters. How can a function in ChildClass call ParentClass's DoIt function?**

(a) DoIt();
(b) ParentClass.DoIt();
(c) ParentClass::DoIt();
(d) ChildClass.ParentClass.DoIt();

123) **Assume class Freshman extends class Student and class Student extends class Person. Assume further than you have an array of Person (called people) that contains Person objects, Student objects, and Freshman objects, and each of these classes declares its own toString() function. If people[3] is a Freshman, what do you need to do so that people[3]->toString(); calls class Freshman's toString() function?**

(a) nothing; this will happen automatically
(b) declare the toString() function virtual in the Person class
(c) make an array of Person*
(d) both (b) and (c)
(e) declare the toString() function virtual in the Person, Student, and Freshman classes
(f) both (e) and (c)

124) **A function which calls itself is called:**

(a) reflexive
(b) identical
(c) recursive
(d) autonomous

125) **In writing the type of function mentioned in the previous question, which of the following are key parts:**

(a) starting condition, calling with smaller problem to solve
(b) starting condition, calling with larger problem to solve
(c) calling with smaller problem to solve, ending condition
(d) calling with larger problem to solve, ending condition

126) **Suppose class Parent contains**

```
virtual double DoIt() = 0;
```

and class Child (which extends class Parent) contains

```
double DoIt(string param);
```

and no other function named DoIt.

(a) The program will not compile.
(b) You can instantiate objects of class Child, but not of class Parent.
(c) You cannot instantiate objects of class Child or class Parent.
(d) You can instantiate objects of class Child and class Parent.

127) **Pick the best statement:**

(a) overloading and overriding are essentially synonyms

(b) overloading occurs in the same class; overriding
occurs at different levels in an inheritance hierarchy

(c) overriding occurs in the same class; overloading
occurs at different levels in an inheritance hierarchy

(d) different signatures are needed for both overloading and overriding

128) **What are the primary advantages of using the string stream classes?**

(a) being able to use the familiar >> and << operators
(b) being able to convert C strings to C++ string objects
(c) being able to convert C++ string objects to C strings
(d) being able to encrypt files (for security purposes)

129) **What is a random access file?**

(a) a file allocated from the heap, whose location
may be different each time the file is allocated

(b) a file of an arbitrary size, typically
not known until run time

(c) a file with which a program can read
from and/or write to arbitrary locations
(which need not be sequential)

(d) a file than can be accessed at any time, provided
only that the file has been opened