



# 2013 Academic Challenge

## COMPUTER SCIENCE TEST - STATE FINAL

This Test Consists of 30 Questions

### Computer Science Test Production Team

Jim Feher, McKendree University – Author/Team Leader  
Nathan White, McKendree University – Author  
Scott Elliott, John A. Logan College – Reviewer  
Mary Weaver, WYSE – Coordinator of Test Production

### GENERAL DIRECTIONS

Please read the following instructions carefully. This is a timed test; any instructions from the test supervisor should be followed promptly.

The test supervisor will give instructions for filling in any necessary information on the answer sheet. Most Academic Challenge sites will ask you to indicate your answer to each question by marking an oval that corresponds to the correct answer for that question. Only one oval should be marked to answer each question. Multiple ovals will automatically be graded as incorrect answers.

Be sure ovals are marked as  , not  ,  ,  , etc.

If you wish to change an answer, erase your first mark completely before marking your new choice.

You are advised to use your time effectively and to work as rapidly as you can without losing accuracy. Do not waste your time on questions that seem too difficult for you. Go on to the other questions, and then come back to the difficult ones later if time remains.

**\*\*\* TIME: 40 MINUTES \*\*\***

**DO NOT OPEN TEST BOOKLET UNTIL YOU ARE TOLD TO DO SO!**

© 2013 Worldwide Youth in Science and Engineering

“WYSE”, “Worldwide Youth in Science and Engineering” and the “WYSE Design” are service marks of and this work is the Copyright © 2013 of the Board of Trustees of the University of Illinois at Urbana - Champaign.

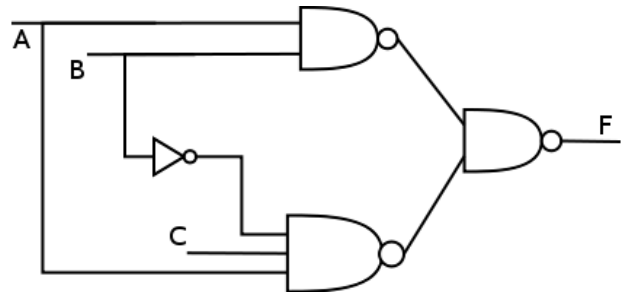
All rights reserved

WYSE – Academic Challenge  
Computer Science Test (State Final) – 2013

1. What is the following 8-bit 2's complement number, 10110101, converted to decimal?
  - a. B5
  - b. 181
  - c. -53
  - d. -181
  - e. None of the above.
2. A floating-point number is comprised of a sign bit, an exponent, and a significand (sometimes referred to as the mantissa). The significand is used to determine the actual digits in the number. If the significand uses 14 binary digits, what is the maximum number of significant decimal digits that can be accurately represented?
  - a. 14
  - b.  $2^{14}$
  - c. 4
  - d. 5
  - e. None of the above.

3. What is the equivalent expression for this circuit?

- a. TRUE
- b. FALSE
- c.  $AB \text{ AND } AB'C$
- d.  $(A'B' \text{ AND } B'CA)'$
- e.  $AB \text{ OR } AB'C$



4. Some of the states in a logical expression never occur, for these states, the output does not matter. These states are often called “don't care” conditions and in this map are listed as D. Using the “don't care” conditions, what is the minimal sum of products expression from the following Karnaugh Map?

- a.  $A'BC \text{ or } AC'$
- b.  $B \text{ or } C'$
- c.  $B \text{ or } AB'C'$
- d.  $BC \text{ or } AC'$
- e.  $B \text{ or } AC'$

|      | A'B' | A'B | AB | AB' |
|------|------|-----|----|-----|
| C'D' | F    | D   | T  | T   |
| C'D  | F    | D   | T  | T   |
| CD   | F    | T   | D  | F   |
| CD'  | F    | T   | D  | F   |

5. Within a relational database, which of the following best describes a foreign key?
  - a. It uniquely identifies a record.
  - b. It is an attribute of one table that is a primary key within another table.
  - c. It is used to translate the information in the database to another language.
  - d. It checks for duplicates within a record.
  - e. It ensures that databases are always normalized.

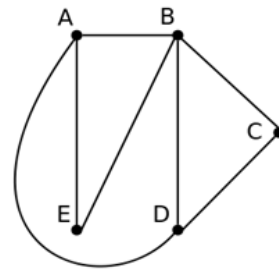
Use the following code for Questions 6 through 9.

```

1  class Ball {
2      public:
3          Ball(float=1);
4          virtual float bounce(float) const;
5          float getRadius() const { return radius; }
6          void setRadius(float r) { radius = (r>0) ? r : 1; }
7          bool operator > (Ball &right) const { return radius > right.radius; }
8      private:
9          float radius;
10 };
11 Ball::Ball(float r) {
12     setRadius(r);
13 } // end Ball constructor
14
15 float Ball::bounce(float height) const {
16     float newheight = 0;
17     if (height > 0) newheight = height * 0.75; // bounces 3/4 of height
18     return newheight;
19 } // end bounce method
20
21 class Basketball: public Ball {
22     public:
23         Basketball() { setRadius(4.65); } // inches
24         virtual float bounce (float) const;
25 };
26
27 float Basketball::bounce(float height) const {
28     float newheight = 0;
29     if (height > 0) newheight = height * 0.8; // bouncier ball
30     return newheight;
31 } // end Basketball bounce method
32
33 class Baseball: private Ball {
34     public:
35         Baseball() { setRadius(1.45); } // inches
36         virtual float bounce (float) const;
37 };
38
39 float Baseball::bounce(float height) const {
40     float newheight = 0;
41     if (height > 0) newheight = height * 0.2; // not very bouncy
42     return newheight;
43 } // end Baseball bounce method
44
45 Baseball a;
46 Ball b;
47 Basketball c;

```

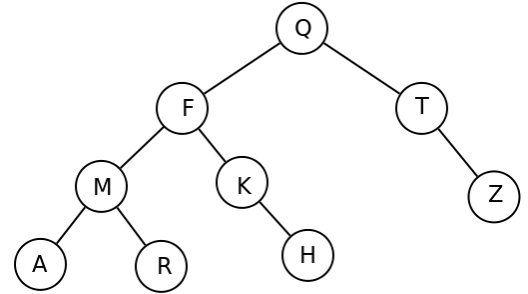
6. In what order are the constructors executed for the code on lines 45 through 47?
- Baseball constructor, Ball constructor, Basketball constructor
  - Ball constructor, Basketball constructor, Baseball constructor
  - Ball constructor, Baseball constructor, Ball constructor, Ball constructor, Basketball constructor
  - Baseball constructor, Ball constructor, Ball constructor, Ball constructor, Basketball constructor
  - No constructors are called for this code.
7. The use of the virtual bounce methods in the classes is an example of which of the following?
- Polymorphism
  - Inheritance
  - Abstraction
  - Encapsulation
  - Static methods
8. Which methods from Ball are available for objects a and c (from lines 45 and 47) to use?
- setRadius, getRadius, overloaded >, bounce for both a and c
  - setRadius, getRadius, overloaded > for both a and c
  - setRadius, getRadius, overloaded > for c and bounce for a
  - setRadius, getRadius, overloaded >, bounce for c
  - setRadius, getRadius, overloaded > for c
9. Which code will set the radius of the Basketball c from line 47 to 3 and then bounce it from a height of 50?
- `c.radius = 3;`  
`c.bounce = 50;`
  - `setRadius(3);`  
`bounce(50);`
  - `c->setRadius(3);`  
`c->bounce(50);`
  - `c.setRadius(3);`  
`c.bounce(50);`
  - No access is allowed to set the radius of a Basketball.
10. A depth first search of the graph starting at node B will search the nodes in which order?  
Assume that given two different nodes, the node with a lower alphabetical placement will be given priority.



- BADEC
- BACDE
- BCDAE
- BADCE
- BAEDC

11. Which is not true regarding the following tree?

- a. The tree has a depth of 3.
- b. The tree has 6 leaves.
- c. This is not a binary search tree.
- d. The tree has a root of Q.
- e. A post-order traversal of the tree would process node A first.



12. A pre-order traversal of the tree from the previous problem would process the nodes in which order?

- a. QFMARKHTZ    b. QFTMKZARH    c. AMRFKHQTZ    d. ARMHKFZTQ    e. QFMARKHZT

13. Which of the following is a client-side, web technology that is primarily used to validate input, create complex user interfaces, and animate web pages?

- a. JavaScript    b. PHP    c. C++    d. HTML    e. None of the above.

14. Which of the following storage mediums is not solid state?

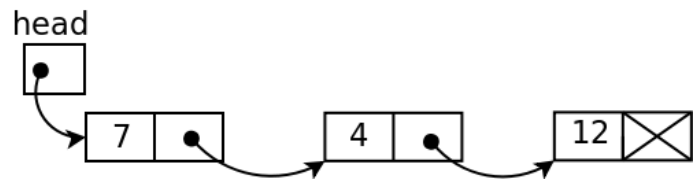
- a. Hard disk drive    b. SD card    c. Flash drive    d. ROM    e. RAM

Use the following code and assume that the linked list provided in the diagram below have already been created when answering questions 15 and 16.

```

typedef struct node* nodeRef;
struct node {
    int data;
    nodeRef next;
};

nodeRef head = new node, curNode;
int sum = 0;
  
```



15. Which of the following code segments adds up and displays the sum of the nodes in the linked list?

- a. 

```
curNode = head;
while (curNode->next != NULL) {
    sum += curNode;
    curNode = curNode->next;
}
cout << sum;
```
- b. 

```
curNode = head;
while (curNode.next != NULL) {
    sum += curNode.data;
    curNode = curNode.next;
}
cout << sum;
```
- c. 

```
curNode = head;
while (curNode != NULL) {
    sum += curNode.data;
    curNode = curNode.next;
}
cout << sum;
```
- d. 

```
for(curNode = head;
    curNode != NULL;
    curNode = curNode->next)
    sum += curNode->data;
cout << sum;
```
- e. All of the segments work properly.

16. Which code segment creates a new node with a data element of 3 and puts it at the front of the list while leaving other three elements at the end of the list?

- a. `curNode = head;`  
`curNode = 3;`  
`head = curNode;`
- b. `curNode = head;`  
`curNode->data = 3;`  
`head = curNode;`
- c. `curNode = new node;`  
`curNode->data = 3;`  
`curNode->next = head;`  
`head = curNode`
- d. `curNode = head;`  
`head = new node;`  
`head->data = 3;`  
`head->next = curNode->next;`
- e. None of the segments work properly.

17. What is the primary purpose of a router?

- a. To compress music files for quicker transmission.
- b. To forward data packets between networks.
- c. It is a mechanism to hide the data, internal structure, and implementation details of a class.
- d. To defragment hard disks to improve performance.
- e. It receives and demodulates data from high frequency tones from DSL lines.

Use the following code for Questions 18, 19, and 20.

```

1 void p1(int& n1)
2 {
3     n1 += 5;
4 }
5 int main()
6 {
7     int tot = 0;
8     for (int i=0;i<13;i+=2) {
9         tot += i;
10        if (i%3==1)
11            p1(tot);
12    }
13    return 0;
14 }
```

18. What is the value of tot when line 13 is reached?

- a. 42
- b. 12
- c. 52
- d. 21
- e. 2

19. How many times will p1 be called?

- a. 20
- b. 19
- c. 3
- d. 2
- e. None of the above.

20. How is the parameter n1 passed to the function p1?

- a. As an integer
- b. By reference
- c. As a constant
- d. By value
- e. None of the above.

Use the following code for Questions 21 through 24.

```

1  int my_func(int n)
2  {
3      switch (n%4) {
4          case 1:
5              n+=n;
6              break;
7          case 2:
8              n*=2;
9          case 3:
10             n*=n;
11             break;
12          case 4:
13             n=1;
14             break;
15          default:
16             n=0;
17      }
18      return n;
19  }
```

21. Which one of the function calls will result in lines 13 & 14 executing?

- a. my\_func(2)    b. my\_func(3)    c. my\_func(0)    d. my\_func(4)    e. None of the above.

22. What is the purpose of break on lines 6, 11, and 14?

- a. It transfers execution to the statement after the conclusion of the switch block.  
 b. It causes the function to return a null value.  
 c. It changes the value of n to the value in the next case statement.  
 d. It causes the switch statement to loop until (n%4) is true.  
 e. It has no purpose other than to help document the code.

23. What will be returned for function call, my\_func(13)?

- a. 26                      b. 16                      c. 10                      d. 49                      e. 2

24. What will be returned for function call, my\_func(2)?

- a. 26                      b. 16                      c. 10                      d. 49                      e. 2

Use the following code for Questions 25 & 26.

```

1  char a;
2  cin >> a;
3  if ((a >= 'A') && (a <= 'Z'))
4      cout << "it worked";
```

25. What is the operator, &&, on line 3?

- a. Logical OR    b. Logical XOR    c. Logical Negation    d. Logical AND    e. Logical NOR

26. Line 4 would execute if the variable a were set to which of the following values?

- a. 'z'                      b. 0                      c. 24                      d. 'G'                      e. 'd'

Use the following code for Questions 27 & 28.

```
1  c = (a>b) ? a : b;
2  cout << c;
```

27. If the variable a was set to 4 and the variable b was set to 8, what would be output when line 2 executes?

- a. 4                      b. 12                      c. 8                      d. 0                      e. None of the above

28. What is the operator, ? :, on line 1?

- a. Logical AND    b. Type Casting    c. Bitwise AND    d. Not Equal    e. Conditional

Use the following code for Questions 29 & 30.

```
1  void printarray (int arg[], int length) {
2      for (int n=0; n<length; n++)
3          cout << arg[n] << " ";
4      cout << endl;
5  }
6  void changearray(int arg[], int a, int b) {
7      arg[a] = b;
8  }
9  int main() {
10     int myarray[] = {5, 10, 15, 20};
11     printarray (myarray,4);
12     changearray (myarray,1,4);
13     printarray (myarray,4);
14     return 0;
15 }
```

29. What is output to the screen as a result of the function call on line 13?

- a. 5 10 15 20    b. 5 4 15 20    c. 5 1 15 20    d. 4 10 15 20    e. 5 10 1 20

30. How is myarray passed to the printarray and changearray functions?

- a. It is passed by value.  
b. An entire copy of the array is duplicated and passed to the functions.  
c. The array is passed as a global variable.  
d. A pointer to the array is passed to the function.  
e. A pointer to each array entry is passed by reference.