You may take this test with you after the test, but you must turn in your answer sheet.

This test has 20 multiple-choice questions, each worth 5 points, for a total of 100 points. This test is worth 15% of your final grade. You must put your answers on the bubble form.

This test is open book and open notes. For the multiple choice problems, select the best answer for each one and select the appropriate letter on your answer sheet. Be careful - more than one answer may seem to be correct. Some questions are tricky. When a problem describes a segment or fragment of code you may assume the rest of the program is correct and would be supplied to make it work. Assume all code is in a C++ program compiled with a C++ compiler.

1. Consider the program shown at right below used to find the average of input numbers. Values to be averaged are typed in, followed by -1 to indicate the end of the input values. Which line could be either added somewhere in the program or could be used to replace a current line in the program to make this program work correctly?

A) count = 1;
B) sum = sum + input;
C) int answer=1.0*(sum/count);
D) if( input == -1) break;

```c
int sum, count, input;
sum=count=input=0;
printf("Provide numbers then -1 \n");
while( input !=-1) {
    scanf("%d", &input);
    sum += input;
    count++;
}
int answer = sum / count;
printf("Average is %d", answer);
```

2. Consider the code segment shown below. If after the function call to changeLetters(...) the value of number has changed, what is the most likely cause?

```c
int number = 5;
printf( " %d", number);
char letters[]="ABCD";
changeLetters( letters);
printf(" %d", number);
```

a) number is a global variable instead of a local variable as it should be
b) Although number is not passed to function changeLetters(), function changeLetters() itself calls a second function which changes number
c) There is some ASCII control characters that are present in the code even though they are not visible
d) Function changeLetters() overwrites the end of array letters
3. What is the output from the code segment shown at right below, called with `start();`

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A) 4</td>
<td>int x = 3; // global variable</td>
</tr>
<tr>
<td>B) 5</td>
<td>void s1( int y)</td>
</tr>
<tr>
<td>C) 6</td>
<td>{ printf(&quot;%d&quot;, x+y); }</td>
</tr>
<tr>
<td>D) 7</td>
<td>void s2( int y)</td>
</tr>
<tr>
<td></td>
<td>{ x = y++;</td>
</tr>
<tr>
<td></td>
<td>s1( y); }</td>
</tr>
<tr>
<td></td>
<td>void start()</td>
</tr>
<tr>
<td></td>
<td>{ x = 1;</td>
</tr>
<tr>
<td></td>
<td>s2( ++x); }</td>
</tr>
</tbody>
</table>

4. What is the output of the following C++ program?

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

int s=1, y=3;

void confuse1(int &y, int s)
{ s++; y++; }

void confuse2(int b, int *s)
{ y = ++(*s); *s = b; }

void confuse3(int &a, int *s)
{ a = *s + 1; (*s)++; }

int main()
{ int s=2;

    confuse1( s, y);
    confuse2( s, &y);
    confuse3( s, &y);
    printf("s + y = %d \n", s+y);
    return 0;
}
```

a) s + y = 3  
b) s + y = 6  
c) s + y = 8  
d) s + y = 10
5. What numbers are in the values array after running the code segment shown at right below?
   A) 1 3 4 7 9
   B) 9 7 4 3 1
   C) 1 3 4 3 1
   D) 9 7 4 7 9

   ```cpp
   const int Max = 5;
   int values[Max] = {1,3,4,7,9};
   for (int i=0; i<Max/2; ++i) {
     values[i] = values[Max-i-1];
   }
   ```

6. Consider two functions that have the same name. Consider which of the following must also be true so that there is not a compiler error:
   I. They have different numbers of parameters
   II. They have different types of parameters
   III. They have different return types
   a) I and III must both be true
   b) II and III must both be true
   c) Either I or II must be true
   d) I, II and III must all be true

For the following three problems consider the following array of numbers to be sorted:

```
7 6 12 2 9 4 3
```

7. How many passes through the data will be required when using Bubble Sort to put the elements into ascending order?
   a) 4
   b) 5
   c) 6
   d) 7

8. How many swaps of element pairs will be done when using Bubble Sort to put the elements into ascending order?
   a) 5 to 7
   b) 8 to 9
   c) 10 to 11
   d) 12 or more

9. How many swaps of element pairs will be done when using Selection Sort to put the elements into ascending order?
   a) 5 to 7
   b) 8 to 9
   c) 10 to 11
   d) 12 or more

10. Consider left-shifting a binary number, adding a zero on the right. The resulting value is:
    a) Unaffected
    b) Half of the original
    c) Twice the original
    d) The opposite (odd/even) of the original number
11. Consider trying to find a particular number within an unsorted array of 100 unique random numbers. On average how many numbers will have to be examined before the number you are searching for is found?

a) 6  
b) 7  
c) 10  
d) 50  
e) 100

12. Consider using binary search to find a particular number within a sorted array of 100 unique random numbers. On average how many numbers will have to be examined before the number you are searching for is found?

a) 6  
b) 7  
c) 10  
d) 50  
e) 100

13. Consider the following declaration:

```c
struct Person {  
    char name[25];  
    int age;  
};

Person p1 = {"Abbott", 45};  
Person *pPerson = &p1;
```

In response to the prompt:

```
printf("Enter first name and age: ");
```

which of the following would compile and run, allowing storing first name and age properly?

a) `scanf("%s %d", p1.name, *(pPerson.age) );`

b) `scanf("%s %d", p1.name, (*pPerson).age );`

c) `scanf("%s %d", name.p1, age.(*pPerson) );`

d) `scanf("%s %d", p1.name, &(pPerson->age) );`
14. What is the output from the code segment shown at right below, called with scope();

A) 2
B) 3
C) 4
D) 5

```c
int n=0; // global variable

void t2( int y)
{
    printf("%d", n+y);
}

void t1( int y)
{
    y++;
    int n=y;
    t2( y);
}

void scope()
{
    n=2;
    t1( n);
}
```

15. Consider the two programs shown below:

Option A:
```c
#include <stdio.h>

int main()
{
    int value;
    printf("Enter value: \n");
    scanf("%d", &value);
    printf("+1 is:%d",++value);
    return 0;
}
```

Option B:
```c
#include <iostream>
using namespace std;

int main()
{
    int value;
    cout << "Enter value:";
    cin >> value;
    cout << "+1 is:" << value+1;
    return 0;
}
```

Which of the following is the best answer regarding the above two programs A and B?

a) Neither A nor B will compile and run as expected.
b) A will compile and run as expected, but B will not
c) B will compile and run as expected, but A will not
d) Both A and B will compile and run as expected.
16. What is the minimum number of lines of code shown below that would need to be corrected in order to get the program segment shown below to compile and run correctly?

```c
char age;
printf("Enter years old up to 300: ");
scanf("%2.1f", age);
if( age => 75) {
    printf("Could be an antique ");
else
    printf("Not yet\n");
}
```

a) 2 lines  
b) 3 lines  
c) 4 lines  
d) 5 or more lines

17. Consider the two program segments shown below:

Option A invoked with callc1();
```c
int cl( int *x, int &y)
{
    *x = *x + 1;
    y = y + 1;
    return y + *x;
}
void callc1()
{
    int x=1;
    int y=3;
    printf("%d", cl( &x, &y) );
}
```

Option B invoked with callc2();
```c
int c2( int x, int y)
{
    x = x + 2;
    y = y + 1;
    return y + x;
}
void callc2()
{
    int x=1;
    int y=2;
    printf("%d", c2( &x, y) );
}
```

Which of the following is the best answer regarding the above two programs A and B?

a) Neither A nor B will compile and run as expected.  
b) A will compile and give output of 6, but B will not  
c) B will compile and give output of 6, but A will not  
d) Both A and B will compile and run and give output of 6
18. What is the output of the code segment below when called with:
   
   ```
   char aString[] = "One.Two.Three";
   char *pString = aString;
   f10( pString, '.');
   ```

   a) One Two
   b) Two Three
   c) One.Two Three
   d) One Two.Three

19. What is the output of the code segment shown at right when function `testf1()` is called?

   ```
   void fc( char *pC)
   {
       *pC = '\0';
   }

   void f10( char *aString, char c)
   {
       char *pChar1 = strchr( aString, c);
       fc( pChar1++);
       char *pChar2 = strrchr( pChar1, c);
       fc( pChar2++);
       printf("%s %s\n", pChar1, pChar2);
   }

   void f1( int y)
   {
       static int x=0;
       x++;
       printf("%d ", x);
   }

   void testf1()
   {
       int x = 2;
       for( int x=0; x<3; x++) {
           f1( x);
       }
   }
   ```

   a) 0 0 0
   b) 1 1 1
   c) 0 1 2
   d) 1 2 3

20. Consider the following two code options to dynamically allocate space for an array of integers:

   ```
   Option A: int *pNewArray = new int[size];
   ```

   ```
   Option B: int *pNewArray = (int *) malloc( sizeof(int) * size);
   ```

   Which of the descriptions below of the above two lines is most accurate?

   a) Neither option is correct
   b) Only Option A is correct
   c) Only Option B is correct
   d) Both are correct