Fall Semester 14, Dr. Punch. Exam #2 (11/6), form 2 A

Last name (printed): ______________________________________________________

First name (printed): _____________________________________________________

Directions:

a) DO NOT OPEN YOUR EXAM BOOKLET UNTIL YOU HAVE BEEN TOLD TO BEGIN.
b) You have 80 minutes to complete the exam (10:20-11:40)
c) This exam booklet contains 30 multiple choice questions, each weighted equally (5 points). **6, double-sided, pages total**
d) You may use one 8.5" x 11" note sheet during the exam. No other reference materials or calculating devices may be used during the examination.
e) Questions will not be interpreted during the examination.
f) You should choose the single best alternative for each question, even if you believe that a question is ambiguous or contains a typographic error.
g) Please fill in the requested information at the top of this exam booklet.
h) Use a #2 pencil to encode any information on the OMR form.
i) Please encode the following on the OMR form:
   - Last name and first initial
   - MSU PID
   - Exam form (see the title of this page)
j) Please sign the OMR form.
k) Only answers recorded on your OMR form will be counted for credit.
l) Completely erase any responses on the OMR form that you wish to delete.
m) You must turn in this exam booklet and the OMR form when you have completed the exam. When leaving, please be courteous to those still taking the exam.

Good luck.

**Timing tip.** A rate of 2.5 minutes per multiple choice problem leaves 5 minutes to go over any parts of the exam you might have skipped.
1) What type is $e$ on Line 1 of Figure 1?
   a) vector<long>
   b) long
   c) vector
   d) long&
   e) None of the above.

2) What type is $itr$ on Line 2 of Figure 1?
   a) vector<long>
   b) long
   c) vector
   d) long&
   e) None of the above.
3) What output is produced by Line 3 in Figure 1?
   a) 0  
   b) 1  
   c) 2  
   d) 3  
   e) None of the above.

4) What output is produced by Line 4 in Figure 1?
   a) 2  
   b) 4  
   c) 6  
   d) 8  
   e) None of the above.

5) What output is produced by Line 5 in Figure 1?
   a) 0  
   b) 1  
   c) true  
   d) false  
   e) None of the above.

6) What output is produced by Line 6 in Figure 1?
   a) 0  
   b) 1  
   c) 2  
   d) 6  
   e) None of the above.
7) Which of the following is true about friend functions?
   a) they are declared in a class
   b) a function can request being a friend to a class
   c) they have a variable named this associated with them
   d) All of the above
   e) None of the above

8) Which of the following are true about a lambda construct?
   a) it is a function and can be invoked like a function
   b) it has a name
   c) it cannot be used in a generic algorithm
   d) All of the above
   e) None of the above

9) What is the meaning of the * operator
    a) used to create a pointer variable
    b) used to dereference a pointer variable
    c) used for multiplication
    d) depends on the context
    e) None of the above.

10) What is meaning of the keyword auto in a type declaration?
    a) Determines a variable's type at compile time
    b) Does not create variables as references unless explicitly requested
    c) Does not create const variables unless explicitly requested
    d) All of the above
    e) None of the above

11) Which of the following are true about C++ methods/function members?
    a) When defined outside the class, they require scope resolution operators.
    b) are invoked in the context of a classes' instance
    c) they are automatically provide the this pointer
    d) All of the above
    e) None of the above

12) Which of the following generic algorithms would you use to print all the values of a vector without any modification?
    a) copy
    b) transform
    c) sort
    d) accumulate
    e) None of the above
13) For the program in Figure 2, what type is `p` in Line 1.
   a) long
   b) map<long, string>
   c) map<string, long>
   d) map
   e) None of the above
14) For the program in Figure 2, give the output of Line 2.
   a) abc
   b) cde
   c) 1
   d) 2
   e) None of the above

15) For the program in Figure 2, give the output of Line 3.
   a) 1
   b) 2
   c) 3
   d) 4
   e) None of the above

16) For the program in Figure 2, give the output of Line 4.
   a) abc
   b) bca
   c) cde
   d) ace
   e) None of the above

17) For the program in Figure 2, give the output of Line 5.
   a) 1
   b) 2
   c) 3
   d) 4
   e) None of the above

18) For the program in Figure 2, give the output of Line 6.
   a) 1
   b) 2
   c) 3
   d) 4
   e) None of the above
For the program shown in Figure 3, which is true about `operator<<` on Line 1?

a) it is a function
b) it has access to private data
c) it can chain multiple calls
d) All of the above
e) None of the above
20) For the program shown in Figure 3, what is called by Line 2?
   a) the default constructor
   b) the two-arg constructor
   c) the three-arg constructor with a default on the third arg
   d) not a constructor call
   e) None of the above

21) For the program shown in Figure 3, what output is produced by Line 3?
   a) 100,300
   b) (100,300)
   c) unknown
   d) (100,300,unknown)
   e) None of the above

22) For the program shown in Figure 3, what output is produced by Line 4?
   a) set
   b) unset
   c) unknown
   d) empty string
   e) None of the above
```cpp
#include<iostream>
using std::cout; using std::endl;
#include<deque>
using std::deque;
#include<algorithm>
using std::transform; using std::copy;
#include<initializer_list>
using std::initializer_list;
#include<iterator>
using std::front_inserter;

long f(long l){
    if (l%2)
        return l*l;
    else
        return l+l;
}

class AClass{
private:
    deque<long> d_; 
public:
    AClass(): d_(deque<long>());
    AClass(initializer_list<long>);
    long m1(long);
    long m2(long l) {return d_[l];};
};

AClass::AClass(initializer_list<long> il){
    copy(il.begin(), il.end(), front_inserter(d_));
}

long AClass::m1(long l){
    transform(d_.begin(), d_.begin()+l, d_.begin(), f);
    return l;
}

int main(){
    AClass ac({1,2,3,4,5});
    cout << ac.m2(0) << endl;  // LINE 1
    cout << ac.m1(3) << endl;  // LINE 2
    cout << ac.m2(0) << endl;  // LINE 3
    cout << ac.m2(3) << endl;  // LINE 4
}
```

Figure 4
23) For the program in Figure 4, what value is printed by Line 1?
   a) 1
   b) 2
   c) 3
   d) 4
   e) None of the above

24) For the program in Figure 4, what value is printed by Line 2?
   a) 1
   b) 2
   c) 3
   d) 4
   e) None of the above

25) For the program in Figure 4, what value is printed by Line 3?
   a) 1
   b) 2
   c) 3
   d) 4
   e) None of the above

26) For the program in Figure 4, what value is printed by Line 4?
   a) 1
   b) 2
   c) 3
   d) 4
   e) None of the above
```
#include<iostream>
using std::cout; using std::endl; using std::ostream;
using std::string; using std::to_string;

class SomeClass{
private:
    long l_;
    string s_;
public:
    SomeClass() = default;
    SomeClass(long l, string s): l_(l), s_(s) {}
    SomeClass(long l): l_(l), s_(to_string(l)) {}
    friend SomeClass operator+(SomeClass, SomeClass);
    friend ostream& operator<<(ostream&, SomeClass&);
};

SomeClass operator+(SomeClass s1, SomeClass s2){
    SomeClass result;
    result.l_ = s1.l_ + s2.l_;
    result.s_ = s1.s_ + s2.s_;
    return result;
}

ostream& operator<<(ostream& out, SomeClass& sc){
    out << sc.l_ << " " << sc.s_ << endl;
    return out;
}

int main(){
    SomeClass sc1(12,"10");
    SomeClass sc2;
    cout << sc1 << endl;  // LINE 1
    sc2 = sc1 + sc1;
    cout << sc2 << endl;  // LINE 2
    sc2 = sc1 + 22;
    cout << sc2 << endl;  // LINE 3
}
```

27) For the program in Figure 5, what output is produced by Line 1?
   a) 12
   b) 10
   c) 12 10
   d) 10 12
   e) None of the above
28) For the program in Figure 5, what output is produced by Line 2?
   a) 12 10
   b) 24 20
   c) 24 10
   d) 24 1010
   e) None of the above

29) For the program in Figure 5, what output is produced by Line 3?
   a) 12 10
   b) 34
   c) 12 22
   d) 34 1022
   e) None of the above

30) Is the addition operator commutative (same result regardless of order) for the class SomeClass in Figure 5?
   a) yes, for all C++ types in combination with SomeClass
   b) only for two SomeClass types
   c) only for two long types
   d) no for all types
   e) None of the above