Université d'Ottawa
Faculté de génie

École de science
d'informatique
et de génie électrique

University of Ottawa
Faculty of Engineering

School of Electrical
Engineering
and Computer Science

uOttawa

L'Université canadienne
Canada's university

CSI2372A
Advanced Programming Concepts with C++

MIDTERM EXAMINATION A

Length of Examination: 75 minutes **October 4, 2017, 14:30**

Professor: Jochen Lang **Page 1 of 10**

Family Name: _____

Other Names: _____

Student Number: _____

Signature _____

You are allowed **ONE TEXTBOOK** as a reference. No calculators or other electronic devices are allowed.

Please answer the questions in this booklet. The marks for each question are marked in the booklet with [ ].

At the end of the exam, when time is up: Stop working and turn your exam upside down. Remain silent.

Question	Marks	Maximum
A.1-A.8		8
B.1		4
B.2		5
C.1		2
C.2		3
C.3		4
Total		26

PART A: SHORT QUESTIONS (8 MARKS)

1. Given the following declaration [1]

```
bool copyWord( const char* in, char* out);
```

call the function `copyWord` with the arguments

```
char in[]{"Hello"};
char* out = new char[128]{};
bool result;
```

2. What is printed by the following? [1]

```
std::string s;
s+="1x";
s+='2';
s[1] = '0';
cout << s.c_str() << endl;
```

3. What is printed by the following? [1]

```
double d = 3.0;
int i = 2;
char c = 1;
double r = d/i+c;
cout << r << endl;
```

4. What is printed by the following? [1]

```
class AC {
  int d_i{23};
  int d_j;
public:
  AC( int k=15 ) : d_i{k}, d_j{k} {}
  void print() {
    cout << d_i << " " << d_j << endl;
  }
};

AC ac{10};
ac.print();
```

5. What is printed by the following? [1]

```
int ras = 8;
int rbs = 2;
int rcs = 3;
int rcd = ras >> --rcs << rbs/2;
cout << rcd << endl;
```

6. What is printed by the following? [1]

```
int a[]{2,4,6};
int *pA = &a[0];
int **pB = &pA;
++pA;
cout << **pB << endl;
```

7. Rewrite the following definition of **a** using auto to define exactly the same type: [1]

```
int x=0;
int &a = x;
```

8. Rewrite the following definition of **b** using auto to define exactly the same type: [1]

```
int x=0;
const int *b = &x;
```

PART B: Short Programs (9 MARKS)

1. The following program does not compile. Mark the lines were an error occurs and state the reason for the error [4]

```
using std::cout;
using std::endl;

class Bear {
  int d_age;
public:
  Bear( int age ) : d_age(age) {}
  void print() {
    cout << d_age << endl;
  }
};

class Cave {
  double d_depth{3.0};
  Bear d_b;
public:
  Cave( Bear b ) : d_b{b} { d_b.d_age = 0; }
  bool hasOldBear() {
    if (d_b.d_age>10) return true;
    else return false;
  }
  Bear getBear() { return d_b; }
};

int main() {
  Cave c;
  Bear b;
  if ( c.hasOldBear() ) {
    b = c.getBear();
    b.print();
  } else {
    cout << "Young Bear!" << endl;
  }
  return 0;
}
```

2. Consider the following definition of the function matrix and the corresponding main routine:

```
int* matrix( int nRows, int nCols ) {
  int *numbers = new int[nRows*nCols];
  int *element = numbers;
  for(int r=0; r<nRows; ++r) {
      for(int c=0; c<nCols; ++c) {
         *(element++) = r*nCols+c;
      }
  }
  return numbers;
}

int main() {
  int nRows, nCols;
  cout << "Number of rows: "; cin >> nRows; cout << endl;
  cout << "Number of cols: "; cin >> nCols; cout << endl;
  auto numbers = matrix(nRows,nCols);
  for(int r=0; r<nRows; ++r) {
      for(int c=0; c<nCols; ++c) {
        cout << numbers[r*nCols+c] << " ";
      }
      cout << endl;
  }
  return 0;
}
```

a. What is printed by the program if the user enters 2 as “number of rows” and 3 as “number of columns”? [1]

b. Rewrite the function matrix to have the prototype `std::vector<int> matrix( int nRows, int nCols )`. The main function will still work as is [3].

c. Could you use `std::array` instead of `std::vector` in the redefinition of the function matrix instead? Explain why in one sentence. [1]

PART C: PROGRAMMING QUESTIONS (9 MARKS)

Consider the following main program:

```
#include <iostream>
#include <cmath>

using namespace std;

int main() {
  double a[]{ 2.0, -3.0, 2.0 };
  double b[]{ -1.0, -1.0, 2.0, 1.0 };
  print( a, end(a)-begin(a) );
  print( b, end(b)-begin(b) );
  cout << length(a,end(a)-begin(a)) << endl;
  cout << length(b,end(a)-begin(a)) << endl;
  auto c = append( a, end(a)-begin(a), b, end(b)-begin(b));
  print( c, static_cast<size_t>(end(a)-begin(a)) +
              static_cast<size_t>(end(b)-begin(b)));
  return 0;
}
```

Program Output:

```
2 -3 2
-1 -1 2 1
4.12311
2.44949
2 -3 2 -1 -1 2 1
```

---

1. Implement the function `print` [2].

2. Implement the function `length` that calculates the 2-norm or magnitude/length of a vector. The vector length is defined as $|\vec{x}| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$ [3].

3. Implement the function `append` [4].