Computer Programming (C++) Midterm exam.

Nov. 24, 1999.

1. What is wrong with the following while repetition structure:

```
while ( z >= 0)
    sum += z;
```

ANS: It's an infinite loop.

2. Write down the output generated by the following programs, respectively.

(a)

```
#include <iostream.h>
int main()
{
   int x=10, product=5;
   product *= x++;
   cout << product <<  '\n' << x << endl;
   return 0;
}
```

Ans:

50

11

(b)

```
#include <iostream.h>
int main()
{
   int x= 5, quotient=5;
   quotient /= ++x;
   cout << quotient <<  '\n' << x << endl;
   return 0;
}
```

ANS:

0

6

3.

(a) What are the outputs generated by the following program?

```
#include <iostream.h>
int main()
{
    for (int i=1; i<= 5; i++) {
      for (int j=1; j<= 6; j++) {
        cout <<   '*' ;
      }
      cout << endl;
    }
    return 0;
}
```

ANS:

```
******
******
******
******
******
```

(b) What are the outputs generated by the following program?

```
#include <iostream.h>
int main()
{
    for (int i=1; i<= 5; i++) {
      for (int j=1; j<= i; j++) {
        cout <<   '*' ;
      }
      cout << endl;
    }
    return 0;
}
```

ANS:

```
*
**
***
****
*****
```

4. What are the outputs generated by the following programs?

```
#include <iostream.h>
int main()
{  int x = 9, &y = x;
   cout << "x = " << x << endl << "y = " << y << endl;
   y = 10;
   cout << "x = " << x << endl << "y = " << y << endl;
   return 0;
}
```

ANS:

x=9

y=9

x=10

y=10

5. What are the outputs generated by the following programs?

```
#include <iostream.h>
int squareByValue( int );
void squareByReference( int & );
int main()
{   int x = 3, z = 5;
    cout << squareByValue( x ) << endl;
    cout << "x = " << x << endl;
    cout << "z = " << z << endl;
    squareByReference( z );
    cout << "x = " << x << endl;
    cout << "z = " << z << endl;
    return 0;
}
int squareByValue( int a )
{
   return a *= a;
}

void squareByReference( int &cRef )
{
   cRef *= cRef;
}
```

ANS:
9
x=3
z=5
x=3
z=25

6. Find the error in each of the following program segments and explain how the error can be corrected.

(a)

```
int sum(int x, int y) {
  int result;
  result = x+y;
}
```

(b)

```
void f(float a);
{   float a;
    cout << a << endl;
}
```

ANS:

(a) add the following return statement before }.

```
return result;
```

(b) should be

```
void f(float a)
{
    cout << a << endl;
}
```

7. Can the following function `sum` do the job of adding integer numbers from 1 to n, given n a positive number? Indicate what is wrong if any.

```
int sum(int n) {
  if (n == 0)
    return 0;
  else
    return n + sum(n);
}
```

ANS: The last statement should be

```
return  n + sum(n-1);
```

8.

(a) What does the following function do?

```
int doSomeThing( const char *s)
{
   int x;
   for ( x =0; *s != '\0'; s++)
      ++x;
   return x;
}
```

ANS: String length calculation.

(b) Do you consider the above function can also be defined as following?

```
   int doSomeThing( const char *s)
   {
     for ( int x =0; *s != '\0'; s++)
        ++x;
     return x;
    }
```

ANS: No. It would have a syntax error — x in "return x" statement undefined.

9. (a) Declare an array, named `numbers`, of type `float` with 10 elements, and initialize the elements to the values 0.0, 1.1, 2.2, 3.3, 4.4, 5.5, 6.6, 7.7, 8.8, 9.9. Assume the symbolic constant SIZE has been defined as 10.

ANS: float numbers[SIZE]= {0.0, 1.1, 2.2, 3.3, 4.4, 5.5, 6.6, 7.7, 8.8, 9.9};

(b) Declare a pointer `fPtr` that points to an object of type float and initialize it such that it points to the beginning of the array `numbers`.

ANS: float *fPtr = numbers;

(c) Assign the value 0.0 to numbers[5], using pointer/offset notation with `fPtr`.

ANS: *(fPtr+5) =0.0; // if fPtr have pointed to the beginning of array numbers.

10.

(a) Are the following two declarations with initializer the same?

```
   char vowel[] = "AEIOU";
   char vowel[] = { 'A', 'E', 'I', 'O', 'U'};
```

ANS: the last one should be: `char vowel[] = { 'A', 'E', 'I', 'O', 'U', '\0'};`

(b) Are the following two declarations the same?

```
   int * i, j;
   int *i, *j;
```

ANS: No.