

CSC112 **Spring 2011**
Fundamentals of Computer Science
Practice Problems

1. What is the difference between a local variable and a global variable ?
2. Write a short program that reads numbers from the standard input stream (`cin`) up to end of file and prints the largest and smallest numbers in the set.
3. Write a short program that reads hexadecimal digits up to end-of-line, and prints the equivalent base 10 number.
4. What is the 8-bit base 2 representation of the (base 10) number 37 ?
5. Using two's complement representation, what is the 8-bit representation of the base 10 number -37 ?
6. What does the following program print?

```
#include <iostream>
using namespace std ;
int main()
{
    int a = 12 ;
    cout << 1 + ~a << endl ;
}
```

7. What would the following program print ?

```
#include <iostream>
using namespace std ;
int main()
{
    cout << (25 & -9) << endl ;
}
```

8. The program in problem 7 works, but the one given below does not compile. I.e., `g++` prints an error message. Why ?

```
#include <iostream>
using namespace std ;
int main()
{
    cout << 25 & -9 << endl ;
}
```

Hint: If I name the program file "u.cc", then the error message is:

```
u.cc: In function 'int main()' :
u.cc:5: error: invalid operands of types 'int' and '<unresolved overloaded
function type>' to binary 'operator<<'
```

9. Write a function to accept an array of integers **a**, and the number of integers **n** stored in the array. Your function should return the largest integer in the array. The function header should be:

```
int find_largest( int a[], int n )
```

10. Explain the difference between *pass by value* and *pass by reference* when passing parameters to a C++ function.
11. Write a function to decide if a number is a perfect number. Note: A number is a **perfect** number if the number is the sum of all of its divisors, excluding the number itself. For example, 6 is a perfect number because $1 + 2 + 3 = 6$ and 6 is divisible by each of 1, 2, and 3. The function header is:

```
bool is_perfect( int k )
```

12. Write a function to accept an array of integers **a**, the number of elements in the array **n**, and a target value **x**. The function should return the position where **x** is found in the array **a**. If **x** is not found, then the function should return -1. The function header is:

```
int search( int a[], int n, int x )
```