15-295 (Competition Programming)
Interview Programming Practice Set

Twiddling

- **T1.** Write a function to count the number of bits set in an unsigned int [C/C++]
- **T2.** Write a function to determine if the runtime stack grows up or down [C/C++]
- **T3.** Write a function to determine if the heap grows up or down [C/C++]
- **T4.** Write a function that converts a double to a ratio (numerator and denominator) represent a reduced fraction.
- **T5.** Write a function that converts a double to a ratio (numerator and denominator) represent a reduced fraction – take a third argument, a double, the tolerance. Find the reduced fraction with the smallest denominator within the supplied tolerance of the given double.

Array

- **A1.** Given an array containing n integers, find the sum of the largest m integers.
- **A2.** Given an array containing integers, return the largest sum of contiguous integers in the array
  Example: if the input is (-10, 2, 3, -2, 0, 5, -15), the largest sum is 8
- **A3.** Given an array with integers between 1 and 1,000,000. One integer is in the array twice. Write a method that determines which one as quickly as possible.
- **A4.** Given an array with integers between 1 and 1,000,000. One integer is in the array twice. Write a method that determines which one as quickly as is possible while using as little memory as possible.

String

- **S1.** Given a string containing words, reverse the words
- **S2.** Given a string [StringBuffer in Java] containing words, reverse the words. Do it **in place**
- **S3.** Given a string [StringBuffer in Java] strip out the white space. Do it **in place**.
- **S4.** Given a string, return a new string containing each character of the original only once
- **S5.** Given a string, remove adjacent repeated characters, e.g. "AABBBCDB" → "ABCDB"
- **S6.** Given a string, find the first non-repeating character within the string, e.g. "ABCBDA" → "C"

Linked List

- **L1.** Write a method that determines if a linked list contains a cycle

BST

- **B1.** Write a method to verify that a binary tree is a BST
- **B2.** Write a non-recursive method to print each element of a BST in order

Other Fun

- **O1.** What is the output of the following code

```
void foo(void) {
unsigned int a = 6;
int b = -20;

if ( (a+b) > 0)
        printf ("Greater than 0\n");
else
        printf ("Negative\n");
}
```

O2. Is the code below correct? If not, what is the problem?

```
int square(volatile int *ptr)
{
  return *ptr * *ptr;
}
```

- **O3**. Is this construct legal, and if so what does this code do?

```
int a = 5, b = 7, c;
c = a+++b;
```

References
- http://maxnoy.com/interviews.html
- NAJones@compuserve.com, "The C Test"