Midterm Practice Problems

TA: Brian Choi (schoi@cs.ucla.edu)
Section Webpage: http://www.cs.ucla.edu/~schoi/cs31

ENJOY!

Problem 1

True **False** Variable names can begin with an alphabet or a digit.
True **False** A string cannot be empty.
True False A character cannot be empty.
True **False** An array can have a size of 0.
True **False** `int m = 5.6;` won't compile because types do not match.
True False `int n = 11 / 5;` will create `n` and initialize it with the value of 2.
True False An array index begins with 0.
True False `for (int i = 0; i <= 49; i++)` will run the loop 50 times (assuming `i` is not modified within the loop).
True **False** If there is a function that does not require any parameters, you can omit parentheses when calling it.
True **False** Constant variables can be modified only in the `main()` function.
True **False** `int x = '0';` sets `x` to an integer 0.
True False `int x = 0;` sets `x` to an integer 0.
True False `int x = 0.0;` sets `x` to an integer 0.
True False `int x = 0.5;` sets `x` to an integer 0.
True **True** You will ace the exam.

Problem 2

It returns the number of multiples of 3 less than or equal to k if k is positive, -1 if k is negative.
You can also say that it returns floor(k/3) when k is positive.

Problem 3

If the input is `9`, it compiles successfully and prints `4` on the screen.

If the input is `2`, it becomes an infinite loop! This is a run-time error.

Problem 4

- Wrong inequality. Note that we don't even need this check. (Why not?)

Line 3: `if (num > 0)` to `if (num < 0)`

- A void function cannot return any value.

Line 4: `return -1;` to `return;`
Line 26: `return 0;` to `return;`

- A space gets drawn in front of the last line. For example, if `num == 3`, the last line gets drawn when `i == 2`. But the first for loop runs because `num - i == 1`, drawing a single space.

Line 11: `j < num - i` to `j < num - i - 1`

- Not enough stars drawn at each line.

Line 17: `j < i` to `j < 2*i+1`

Problem 5

Immediately after calling `mystery2`:

In `main`:

a	*b*	*c*
1	2	3

In `mystery2`:

&a	b	&c
b	3	*a*

After the line: `a = b + c;`

In `main`:

a	*b*	*c*
1	4	3

In `mystery2`:

&a	b	&c
b	3	*a*

After the line: `b = a + c;`

In `main`:

a	*b*	*c*
1	4	3

In `mystery2`:

&a	b	&c
b	5	*a*

After the line: `c = a + b;`

In `main`:

a	*b*	*c*
9	4	3

In `mystery2`:

&a	b	&c
b	5	*a*

When `mystery2` terminates, all variables in `mystery2` vanish.

In `main`:

a	b	c
9	4	3

The output is:
9 4 3

Problem 6

```
string rotate(string s, int k)
{
    if (k < 0 || s.empty())
       return s;

    int toRotate = k % s.size();
    return s.substr(s.size() - toRotate) +
           s.substr(0, s.size() - toRotate);
}
```

Problem 7

a)

```
bool isHebrew(string word)
{
   for (int i = 0; i < word.size(); i++)
   {
      if (word[i] == 'a' || word[i] == 'e' || word[i] == 'i' ||
          word[i] == 'o' || word[i] == 'u')
         return false;
   }
   return true;
}
```

b)

```
int hebrew(string words[], int n)
{
   if (n < 0)
      return -1;

   int count = 0;
   for (int i = 0; i < n; i++)
   {
      if (isHebrew(words[i]))
      {
         words[count] = words[i];
         count++;
      }
   }
   return count;
}
```