Sp10 CS31 Disc 1H – Midterm Practice Solutions

TA: Peter Wu (peterhwu@cs.ucla.edu) Webpage: http://www.cs.ucla.edu/~peterhwu/cs31/

1. Pirate and Ninja Talk (hard)

Translate all the occurrences of "HA" (regardless of whether the individual letters are lowercase, uppercase, or a mixture of both) into lowercase and "lol" into uppercase. Examples:
"HAHahaH the pirates aren't very stealthy. Lol!!" should become "hahahaH the pirates aren't very stealthy. LOL!!"
"lollipop hats? no idea. lalalololololo" should become "LOLlipop HAts? no idea. lalaLOLoLOLo"

```
// convert any sequence of "lol" in t
// to uppercase; checks from index i
// Examples: lol -> LOL, lOL -> LOL
bool convertLOL(string& t, const int i)
{
  char c1 = t[i];
  if (toupper(c1) == 'L')
  {
    char c2 = t[i+1];
    if (toupper(c2) == 'O')
    {
      char c3 = t[i+2];
      if (toupper(c3) == 'L')
      {
        t[i+2] = 'L';
        t[i+1] = 'O';
        t[i] = 'L';
        return true;
      }
    }
  }
  return false;
}

// convert any sequence of 'ha' to
// to lowercase; checks text at index i
// Examples: hA -> ha, HA -> ha
bool convertHA(string& text, int i)
{
  char currChar = text[i];
  if (tolower(currChar) == 'h')
  {
    text[i] = 'h';
    i++;
    currChar = text[i];
    if (tolower(currChar) == 'a')
    {
      text[i] = 'a';
      return true;
    }
  }
  return false;
}
// 'text' is input. 'changed' is true
if any
// uppercase/lowercase changes were
made;
// function should return text with the
// changed uppercasing/lowercasing
string changeCase(string text, bool& changed)
{
  string copyText = text;
  changed = false; //what if we removed this?
  for (int i = 0; i < copyText.size(); i++)
  {
    // check if any of them converted
    bool changedLOL = convertLOL(copyText, i);
    bool changedHA = convertHA(copyText, i);
    // save result if either conversion worked
    changed = changedLOL || changedHA;
  }
  return copyText;
}
```

A) Name all types of inputs to changeCase that will cause **undefined behavior/exceptions**, and fix these errors.
- any text that ends in "l", "ol", "h" (either uppercase or lowercase) because convert LOL and convert HA assume that there's an element next without first checking it

B) Assume that part A has been fixed. Something's funny about the convert functions. Name all types of inputs to the convert functions that will cause **incorrect results**, and fix these errors.
convertHA will change all h's to lowercase regardless of whether the h is followed by a 'h,' like "hehe hi" will become "HeHe Hi"

C) Let's assume that the convert functions work as intended, with no funny results. Your pirate friend is wondering why he needs the line marked "//what if we removed this?" in changeCase()?
According to the spec, we should probably return false because we didn't make any changes.

D) Once again, assume that the convert functions work as intended. Name all types of inputs to the changeCase function that will cause an **incorrect result** (either the wrong return value, changedCase.. or both!)

- Won't properly change phrases like "lolol" to "LOLol" (see if you can figure out why? – see if you can fix this)
- Doesn't give the correct value of changed because it's overwritten over every iteration. If changed is true, it should stay true (see if you can fix this)

2. Strength of Ns (hard)

Write a **void** function called 'listNs' that takes in **a nonnegative integer 'min'** and **a nonnegative integer 'max'** with the assumption that **max >= min**, and **prints** out all the values of N^N that lie between min and max, inclusive. N here is a nonnegative int. (Separate each number with a space or newline). You are not allowed to use any special math functions defined in <math> or <cmath>, such as the exponential function. Don't worry about error checking.
Therefore listNs(3,4444) should print: 27 256 3125 because 27 is the smallest N^N (N=3) that falls within the range, and 3125 is the largest N (N=5) that falls within the range. Another example: listNs(0,27) should print: 0 1 4 27. (Hint: either try writing the code that prints out N^2 for all numbers, or write code that computes N^N for a single number)

```
// multiple ways of doing this
void listNs(int min, int max)
{
	for (int currNum = min; currNum < max; currNum++)
	{
		int result = currNum;
		for (int multindex = 1; multindex < currNum; multindex++)
		{
			result *= currNum;
			if (result > max)
				return;
		}
		cout << result << " " << endl;
	}
}
```

3. Switch that statement! (moderate) (try on own)

```
char c = ??;
switch(c)
{
case 'a':
	cout << 1 << endl;
	break;
case 'b':
case 'c':
	cout << 3 << endl;
	break;
case 'd':
	cout << 4 << endl;
case 'e':
	cout << 5 << endl;
	break;
default:
	cout << c << endl;
}
```

What is printed for each of the following: c = 'a' through c = 'f'?

4. Mystery Spell of Code (try on own)

Harry has fallen into a trap set by the evil wizard V, and needs your help to solve the coding problem! What is printed out? Assume proper headers.

```
int lifeleft(int a, int b)
{
	int i;
	for (i=a; i>1 && b>1; i=i/b - 1)
	{
		cout << i << " ";
	}
	cout << "\ndone! a: " << a << endl;
	return i;
}

int iluvpercent(int a, int b)
{
	if (a > b)
		return a %= 5;
	else
		return b %= 5;
}

int main()
{
	int a = 3;
	a = lifeleft(9,a);
	int b = lifeleft(20,1);
	int c = iluvpercent(54,2);
	cout << a << endl;
	cout << b << endl;
	cout << c << endl;
}
```