

CS101 – Computer Programming
Quiz for Wednesday Batch – 15 October 2014

Q1. Consider the following program

<pre>bool compare1(int a, int *x) { return &a == x; } bool compare2(int &a, int *x) { return &a == x; } bool compare3(int &a, int *x) { return a == *x; }</pre>	<pre>int main(){ int a; cin >> a; bool m = compare1(a, &a); bool n = compare2(a, &a); bool p = compare3(a, &a); bool q = m n p; bool r = m && n && p; return 0; }</pre> <p style="margin-top: 20px;">A) Value m is true B) Value n is true C) Value p is true D) Value q is true E) Value r is true F) None of these</p>
---	--

Q2. We want to write a function that takes two character arrays (treated as strings) and determines if they are "same modulo blanks", i.e. contain the same sequence of non-blank characters. Thus, if the first string is " I am fine" and the second is "Iamfine" or even "I amfin e ", we want the function to declare that the two strings are compatible. On the other hand, if the first string is " I am fine" and the second one is "I am Fine", then the function should declare the two strings as incompatible (not the upper case 'F' in one and the lower case 'f' in the other).

Consider the following code snippet for achieving the above purpose. We want the function 'isSameModuloBlanks' to return true if 's1' and 's2' are the same modulo blanks (as described above), and to return false otherwise.

```
bool isSameModuloBlanks(char s1[], char s2[])
{
    int i, j;
    const char c1 = Char1;
    const char blank = ' ';
    const bool b1 = Bool1;

    for (i = 0, j = 0; ((s1[i] != c1) && (s2[j] != c1)); ) {
        // Loop invariant: The prefix of s1 and s2 seen so far are the same modulo blanks.
        while ((s1[i] == blank) && (s1[i] != c1)) { i++; }
        while ((s2[j] == blank) && (s2[j] != c1)) { j++; }

        if ((s1[i] != c1) && (s2[j] != c1) && (s1[i] == s2[j])) {
            // The strings are the same modulo blanks even after examining the next character.
            i++;
            j++;
            continue;
        }
        else if ((s1[i] == c1) && (s2[j] == c1)) {
            return !b1;
            break;
        }
        return b1;
    }
    return true;
}
```

Note that there is an unspecified character constant 'Char1' and an unspecified boolean constant 'Bool1' used in the above code. Which of the following combinations of 'Char1' and 'Bool1' will cause function 'isSameModuloBlanks' to return the correct answer for all strings s1 and s2.

- A) Char1 = '\0' and Bool1 = 'false'
- B) Char1 = ' ' and Bool1 = 'true'
- C) Char1 = '\0' and Bool1 = 'true'
- D) Char1 = ' ' and Bool1 = 'false'
- E) None of these