10. Regular Expression Matching

Implement regular expression matching with support for ‘.’ and
’}’.

‘.’ Matches any single character. ’*’ Matches zero or more of the preceding element.

The matching should cover the entire input string (not partial).

The function prototype should be: bool isMatch(const char \emph{s, const char }p)

Some examples: isMatch(‘‘aa’’,‘‘a’’) → false\\
isMatch(‘‘aa’’,‘‘aa’’) → true\\
isMatch(‘‘aaa’’,‘‘aa’’) → false\\
isMatch(‘‘aa’’, ‘‘a\emph{‘‘) → true\\
isMatch(’’aa‘‘,’’.}’’) → true\\
isMatch(‘‘ab’’, ‘‘.\emph{‘‘) → true\\
isMatch(’’aab‘‘,’’c}a*b‘‘) → true

```
public boolean isMatch(String s, String p) {
    if (p.isEmpty()) {
        return s.isEmpty();
    } else if (p.length() == 1) {
        return s.length() == 1 && isEqual(s, p);
    } else if (p.charAt(1) != '*') {
     return s.length() > 0 && isEqual(s, p) && isMatch(s.substring(1), p.substring(1));
    } else {
        if (s.length() > 0 && isEqual(s, p)) {
        return isMatch(s, p.substring(2)) || isMatch(s.substring(1), p);
        } else {
            return isMatch(s, p.substring(2));
        }
    }
}

private boolean isEqual(String s, String p) {
    return s.charAt(0) == p.charAt(0) || p.charAt(0) == '.';
}
```