

Algorithmics	Student information	Date	Number of session
	UO:		
	Surname:		
	Name:		



## Activity 1. Pattern matching

```

public boolean checkPattern(String pattern) {
    if (pattern.isEmpty()) {
        return emptyPattern(); //O(1)
    }
    if (isEmptyText) {
        table = new boolean[1][pattern.length()];
    } else {
        table = new boolean[text.length()][pattern.length()];
    }

    char letterText = '\0';
    //O(n²)
    for (int i = 0; i < table.length && table != null; i++) { //O(n)
        if (!isEmptyText) {
            letterText = text.charAt(i); //O(1)
        }
        for (int j = 0; j < table[i].length; j++) { //O(n)
            char letterPattern = pattern.charAt(j); //O(1)
            switch (letterPattern) {
                case '?':
                    if (i == 0 || j == 0)
                        table[i][j] = true;
                    else if (table[i - 1][j - 1] || table[i][j - 1])
                        table[i][j] = true;
                    break;
                case '*':
                    if (i == 0 || j == 0)
                        table[i][j] = true;
                    else if (table[i - 1][j - 1] || table[i][j - 1] ||
table[i - 1][j])
                        table[i][j] = true;
                    break;
            }
        }
    }
}

```

Algorithmics	Student information	Date	Number of session
	UO:		
	Surname:		
	Name:		

```

        default: // a normal letter
            if (i == 0 || j == 0)
                table[i][j] = letterPattern == letterText;
            else if (table[i - 1][j - 1])
                table[i][j] = letterPattern == letterText;
            break;
        }
    }
}
return table[table.length - 1][table[0].length - 1] && table[0][0];
}

private boolean emptyPattern() { //O(1)
    table = new boolean[1][1];
    if (text.isEmpty()) { //O(1)
        table[0][0] = true;
        return true;
    }
    table[0][0] = false;
    return false;
}

```

The complexity of the algorithm is  $O(n^2)$

The algorithm could be improved if it didn't calculate the table of partial solutions, and just returns true or false. So we can stop if the first character doesn't match, it can directly return false; and it just goes once through the length instead of 2