

**Jacek Radajewski (43612772)**  
**SCIE2100 Prac 3**  
**14/04/2014**

## Question 1

```
def alignGlobal(seqA, seqB, substMatrix, gap=-1):
    .
    .
    .

    # -----
    # starting at the bottom right hand corner of the matrix trace back to the origin 0,0
    while i > 0 and j > 0:
        # -----
        if S[i, j] == S[i - 1, j] + gap:
            alignA = seqA[i - 1] + alignA
            alignB = "-" + alignB
            i -= 1
        elif S[i, j] == S[i, j - 1] + gap:
            # Got here by a gap in sequence A (go left)
            alignA = "-" + alignA
            alignB = seqB[j - 1] + alignB
            j -= 1
        # -----
        else:
            # Got here by aligning the bases (go diagonally)
            alignA = seqA[i - 1] + alignA
            alignB = seqB[j - 1] + alignB
            i -= 1
            j -= 1
    # Fill in the rest of the alignment if it begins with gaps
    # (i.e., traceback all the way to S[0, 0])
    while i > 0:
        # Go up
        # -----
        alignA = seqA[i - 1] + alignA
        alignB = "-" + alignB
        # -----
        i = i - 1
    while j > 0:
        # Go left
        # -----
        alignA = "-" + alignA
        alignB = seqB[j - 1] + alignB
        # -----
        j = j - 1
    alignment = Alignment([Sequence(alignA, seqA.alphabet, seqA.name, gappy=True), Sequence(alignB, seqB.alphabet, seqB.name, gappy=True)])
    alignment.gap_penalty = gap # save the gap for future analysis
    return alignment
```