AAGB - TME1

1. Needleman & Wunsch

The goal of this exercice is to recode the Needleman & Wunsch algorithm. The different questions are there as a guide to build the algorithm. For this reason, you are not forced to follow them strictly. The algorithm is as follows :

- Initialization : $\begin{cases} S_{i,0} = i \times g \\ S_{0,j} = j \times g \end{cases}$
- Induction : $S_{i,j} = max \begin{cases} S_{i-1,j-1} + \sigma(a_i, b_j) \\ S_{i-1,j} + g \\ S_{i,j-1} + g \end{cases}$
- Traceback : At each step, we keep track of the alignment that gives the best score. This will allow to find the best global alignment(s).

1. A distance matrix D is a square matrix of size n, wher n is the length of the alphabet. Given two letters of that alphabet, $D(A, B)$ corresponds to the distance between A and B. In our case, the alphabet can for example be the set of nuclotides (A,C,G,T), or the twenty amin-acids. We can see the diagonal of D as matches, and the rest as mismatches. For example, for a match value of 1 and a mismatch of -1, D would have ones on the diagonal, and -1 elsewhere. Write a function returning the distance matrix, given an alphabet , a match and a mismatch score.
2. Code a function that returns the score matrix (as seen in the TD) with the Needleman & Wunsch algorithm (without traceback). This function can for instance take the two sequences to align, the distance matrix and a value corresponding to the gap penalty. What is this algorithm's complexity?
3. Add the traceback to have a complete function. The output should be the alignment, and it's associated score. Is this alignment necessarily unique?
4. In general, we consider that opening a gap is less costly than extending a gap. Modify your function to take two values, one for a gap opening, an one for a gap extension.
5. Test this function, for example on the TD sequences : A = (CATGAC) and B = (TCTGAAC).

2. A small example

We would like to see what this give on a real example with proteins. Let's take proteins 2ABL et 1OPK, whose sequences are given here :

- >2ABL:A|PDBID|CHAIN|SEQUENCE

MGPSENDPNLFVALYDFVASGDNTLSITKGEKLRVLGYNHNGEWCEAQ
TKNGQGWVPSNYITPVNSLEKHSWYHGPVSRN AAEYLLSSGINGSF
LVRESESSPGQRSISLRYEGRVYHYRINTASDGKLYVSSESRFNTLAELV
HHHSTVADGLITTLHYPAP

- >1OPK:A|PDBID|CHAIN|SEQUENCE

GAMDPSEALQRPVASDFEPQGLSEAARWNSKENLLAGPSENDPNLFV
ALYDFVASGDNTLSITKGEKLRVLGYNHNGEWCEAQTKNGQGW
VPSNYITPVNSLEKHSWYHGPVSRNAAEYLLSSGINGSFLVRESE
SSPGQRSISLRYEGRVYHYRINTASDGKLYVSSESRFNTLAELVHHHST
VADGLITTLHYPAPKRNKPTIYGVSPNYDKWEMERTDITMKHKLGGG
QYGEVYEGVWKKYSLTVAVKTLKEDTMEVEEFLKEAAVMKEIKHPNL
VQLLGVCTREPPFYIITEFMTYGNLLDYLRECNRQEVSAVVLLYMATQIS
SAMEYLEKKNFIHRNLAARNCLVGENHLVKVADFGLSRLMTGDTYTAH
AGAKFPIKWTAPESLAYNKFSIKS DVWAFGVLLWEIATYGMSPYPGIDL
SQVYELLEKDYRMERPEGCPEKVYELMRACWQWNPSDRPSFAEIHQAF
ETMFQES SISDEVEKELGKRGT

1. Use the BLOSUM62 matrix as distance matrix, with a gap opening cost of 11 and extension of 1, and align the two sequences with Needleman Wunsch.
2. Compare that result with the Needleman Wunsch given by blast.
3. Now we would like to compare these protein structures on the Protein Data Bank. What can we see?
4. On which agorithm does BLAST rely?

2. Smith-Waterman (Bonus)

Needleman & Wunsch gives a global alignement. It is sometimes useful to rather highlight local alignments. In order to do this, reuse the previous algorithm with the following modification:

- Induction : $S_{i,j} = max \begin{cases} 0 \\ S_{i-1,j-1} + \sigma(a_i, b_j) \\ S_{i-1,j} + g \\ S_{i,j-1} + g \end{cases}$
- For the traceback, instead of beginning at the end of the sequece, we start from the highest scoring position.

1. Code this algorithm! It should also give the position of the local alignment. Achtung! there can be several local maxima: you can either give one, or all the best alignements.
2. Compare what you get for the two algorithms...