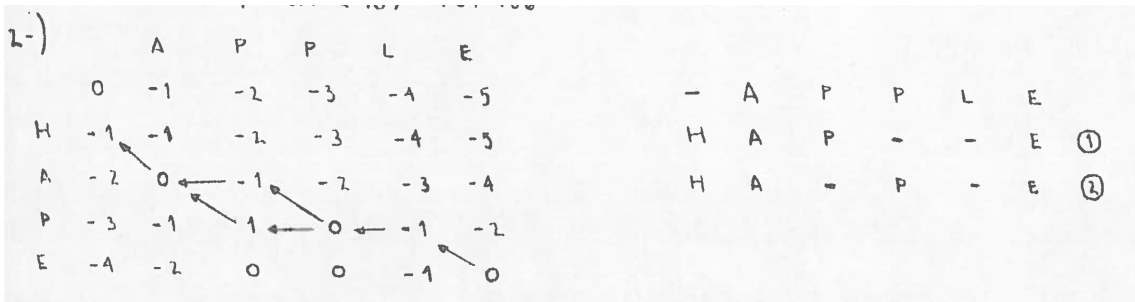
Question 1:



Question 2:

```
S= CCCC
T = CACACAC

Best alignment with linear gap penalty:
C — C — C — C - C
C A C A C A C A C
```

Score with linear gap penalty = +1
Score with affine gap penalty = -5

Best alignment with affine gap penalty:

```
C C C C C - - - -
C A C A C A C A C
```

Score with linear gap penalty = -3
Score with affine gap penalty = -3

Question 3.

a) The two sequences cannot have lengths that are too different, as this would force us to use two consecutive gaps somewhere. In fact, a solution is only achievable if |m-n| <= min(m,n)+1. That's because |m-n| is the number of gaps that will need to be inserted in the shorter sequence. Those gaps have to be interleaved with nucleotides. Starting and ending with a gap, we get min(m,n)+1.

Example for m=7, n=3

```
AAAAAAA
-A-A-A-
```

b) We first observe that the no-multi-gap alignment is a special case of the pairwise alignment problem with affine gap penalty, when the gap extension penalty is infinite: score(L) = -d - e*(L-1), where e = +inf. (Note: this is slightly different from the affine penalty scoring scheme presented in class, which was cost(L) = a + b*L, but it doesn't make a big difference).

Following the algorithm presented in the Durbin et al. book (Equation 2.16), we introduce three dynamic programming tables: M, Ix, and Iy. M is computed with no change. Since e=+inf, Ix(i,j) and Iy(i,j) reduce to just M(i-1,j)-d and M(i,j-1)-d (respectively). Initialization is done as follows (not detailed in the book):
M(0,0)= 0
M(i,0) = M(j,0) = +inf, for i,j>0

Ix(0,0)=0
Ix(1,0) = -d
Ix(i,0) = -inf for i>1

Iy(0,0)=0
Iy(0,1) = -d
Iy(0,j) = -inf for j>1

The rest of the algorithm proceeds like described in the book. Trace-back is performed from max(M(m,n), Ix(m,n), Iy(m,n)).

c) I'm too old for this…

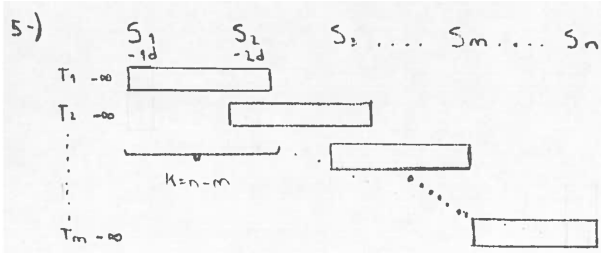
Question 4:

4-) Simply not allow for mismatches by making the penalty of a mismatch $-\infty$
or
Change the recurrence

$$A_{i,j} = \max \begin{cases} A_{i-1,j-1} + 1 & \text{if } S_i = T_j \\ A_{i,j-1} \\ A_{i-1,j} \end{cases} \qquad 0 \text{ if } i=0 \text{ or } j=0$$

Question 5:

5-)



$$F(i,j)\ \forall i,j\ (i-j \geq K) = \text{MAX} \begin{cases} F(i-1,j-1) + s(x_i, y_j) \\ F(i-1,j) - d \end{cases}$$

Basis $F_{0j} = d * j$
$F_{i0} = -\infty$

Question 6:

#1) S_1: TGC

S_2: TCA

S_3: TGCA

First align S_1 and S_2

the best alignment is
```
TGC
TCA
```
score: -1

then align
```
T G C
T C A
```
with TGCA

the best alignment is
```
T - G C
T - C A
T G C A
```
score = 2-4+0+0 = -2

or
```
T G C -
T C A -
T G C A
```
score = 2+0+0-4 = -2

and the final score of the alignment is:
```
T - G C
T - C A
T G C A
```
score: 3-4-1-1 = -3

But there exist a better alignment:
```
T G C -
T - C A
T G C A
```
score 3-3+3-3 = 0

Question 7:

```
S = A A A A A A A A A A A A A A A A A A A A
T = A A A A C A A A A C A A A A C A A A A C
```

These two sequences are 80% identical but contain no exact match of size 5.

Bonus Question:

This can be done using the Hirschberg algorithm, which is described here.
https://en.wikipedia.org/wiki/Hirschberg%27s_algorithm