

Nota inicial: para a resolução da lista de exercícios, foi utilizado Python 3 - com auxílio da biblioteca Bio (para utilização da matriz blosum62 que em termos de TAD é disponibilizada num dicionário).

Questão 1-a:

Pseudocódigo

Foi utilizado o editor VS Code pela facilidade de formatação que inclui também o número de linhas.

```
1 // Primeiro definimos a classe responsável pelas operações de alinhamento
2
3 Classe: Alinhador
4 Recebe os parâmetros:
5     seq1 - sequência 1
6     seq2 - sequência 2
7     penalidadeGap
8     penalidadeMiss
9     scoreDeMatch
10
11 Construtor:
12     Todos os parâmetros viram atributo do objeto, além dos atributos
13     * scoreFinal // que começa com 0
14     * identidade // que começa com 0
15     * matrizAlinhamento // matriz de zeros com (n x m) = (len(seq1)+1, len(seq2)+1)
16     * matrizDeTraceBack // matriz de zeros com (n x m) = (len(seq1)+1, len(seq2)+1)
17     // a matriz de trace back será preenchida com caracteres indicando como foi construída
18     // a matriz de alinhamento. Onde d: diagonal; l: esquerda; u: cima; f: posição (0,0)
19     * string1 // Resultado de alinhamento da seq1
20     * string2 // Resultado de alinhamento da seq2
21
22 Métodos:
23
24     pegaValor(i, j) // recebe uma posição i,j na matrix e compara a posição i-1,j-1
25                     // das sequências, retornando o máximo valor de acordo com a fórmula dada
26                     // além da operação escolhida (match/mismatch, gap na seq1, gap na seq2)
27
28
```

```

29     alinha():
30         Preenche a coluna 0 da matrizDeTraceBack com 'u'
31         Preenche a linha 0 da matrizDeTraceBack com 'l'
32         Assina a posição (0, 0) da matrizDeTraceBack com 'f'
33
34         Preenche a coluna 0 da matrizAlinhamento com valor da linha anterior + penalidadeGap
35         Preenche a linha 0 da matrizAlinhamento com valor da coluna anterior + penalidadeGap
36         Assina a posição (0, 0) da matrizDeTraceBack com 0
37
38         Para i, j da matrizAlinhamento:
39             Se i ou j forem iguais a 0:
40                 Pula iteração // pois já foram preenchidos
41
42                 matrizAlinhamento[i][j], operação = pegaValor(i, j)
43                 matrizDeTraceBack[i][j] = operação
44
45         scoreFinal = última i,j da matrizAlinhamento
46         fazAlinhamento()
47
48
49     fazAlinhamento():
50         string1 = ''
51         string2 = ''
52
53         OperaçãoAtual = matrizDeTraceBack[último i][último j]
54         i,j = últimos i,j da matrizDeTraceBack
55

```

```

56     Enquanto OperaçãoAtual != 'f':
57
58         Se OperaçãoAtual for 'd' (diagonal):
59             string1 recebe o caractere i da seq1    // ou seja, alinha
60             string2 recebe o caractere j da seq2    // ou seja, alinha
61             i = i-1
62             j = j-1
63
64         Se OperaçãoAtual for 'l' (esquerda):
65             string1 recebe -    // ou seja, recebe um gap
66             string2 recebe o caractere j da seq2    // ou seja, alinha
67             j = j-1
68
69         Se OperaçãoAtual for 'u' (cima):
70             string1 recebe o caractere i da seq1    // ou seja, alinha
71             string2 recebe -    // ou seja, recebe um gap
72             i = i-1
73
74         calculaIdentidade()
75
76     calculaIdentidade():
77         Conta o número n da maior string final entre as duas alinhadas
78         identidade =    número de caracteres iguais nas mesmas posições das
79         |               |               |               |               |
80         |               |               |               |               |
81         |               |               |               |               |
82         |               |               |               |               |
83         // Com a classe definida, podemos prosseguir com os alinhamentos
84

```

```

85     gap = -4
86     match = 5
87     mismatch = -3
88
89     scores = {}    // dicionário do tipo "humano vs animal": (score, identidade)
90
91     Para cada animal:
92         alinhador = Alinhador(seqHumana, seqAnimalAtual, gap, mismatch, match)
93         alinhador.alinha()
94         scores["humano vs animalAtual"] = (alinhador.scoreFinal, alinhador.identidade)
95
96     Dentre todos os alinhamentos realizados:
97     Pega (mostra) o com maior valor de score final
98     Se tiver outro com mesmo valor, desempata com a identidade
99     Se ambos score e identidade forem iguais, lista todos estes melhores alinhamentos que empataram

```

Considerações: ao realizar o traceback, o algoritmo de Needleman-Wunsch não pode simplesmente pegar o maior valor dentre diagonal, esquerda e cima, pois para atingir o valor da célula que se está analisando, pode ser que o algoritmo tenha escolhido introduzir um gap mesmo que fosse mais custoso que um mismatch para poder efetuar o matching posteriormente. Ou seja, no traceback não se pode apenas avaliar os valores anteriores e andar para o maior. Por isso, deve-se manter um registro de que operações foram feitas para alcançar o valor de cada célula e podermos efetuar o traceback corretamente. Qualquer Needleman-Wunsch online compactua com essa implementação, e

um exemplo prático mostrando que não funciona simplesmente pegar o maior valor anterior dentre diagonal, esquerda e cima, pode ser visto no link seguinte:

<http://rna.informatik.uni-freiburg.de/Teaching/index.jsp?toolName=Needleman-Wunsch>

Notar que $C_4 \times C_3$ que está em 1 deveria andar para cima cujo valor é 1, no entanto anda na diagonal cujo valor é 0, pois isso possibilita a configuração de alinhamento final que realmente resultou naquele score final de 2.

Resultados:

```
C:\Users\Pichau\Desktop\Biocomp\list 2\part 1>python e2-1a.py
Results for human vs chicken:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 -3 -7 ... -547 -551 -555]
 [-8 -7 2 ... -538 -542 -546]
 ...
 [-552 -543 -534 ... 243 239 235]
 [-556 -547 -538 ... 252 248 244]
 [-560 -551 -542 ... 248 257 253]]

VLSPA-DKTNVKAAGKVGGA-HAGEYGAEALERMLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALS--A-LSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLAS-VSTVLTISKY-
ML-TAEDKKLIQQAWEK-AASHQEEFGAEALTRMFTTYPQTKTYFPHFDLSPGSDQVRGHGKVKLGALGNVKNV-D--N-LSQAMAELSNLHAYNLRVDPVNFKLLSQCIQVVLAVHMGKDYTPEVHAADFDFL-SAVSAVLAEKYR

Final Score: 253
Identity: 0.6312056737588653

Results for human vs cow:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 5 1 ... -547 -551 -555]
 [-8 1 10 ... -538 -542 -546]
 ...
 [-552 -543 -534 ... 550 546 542]
 [-556 -547 -538 ... 559 555 551]
 [-560 -551 -542 ... 555 564 560]]

VLSPADKTNVKAAGKVGGAHAGEYGAEALERMLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTISKY-
VLSAADKGNVKAAGKVGGAHAGEYGAEALERMLSFPTTKTYFPHFDLSHGSAQVKGHGKAAALTKAVEHLDLPGALSELSDLHAHKLRVDPVNFKLLSHSLLVTLASHLPDFTPAVHASLDKFLANVSTVLTISKYR

Final Score: 560
Identity: 0.8723404255319149
```

```
Results for human vs deer:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 5 1 ... -547 -551 -555]
 [-8 1 10 ... -538 -542 -546]
 ...
 [-552 -543 -534 ... 430 426 422]
 [-556 -547 -538 ... 439 435 431]
 [-560 -551 -542 ... 435 444 440]]

VLSPADKTNVKAAGKVGGAHAGEYGAEALERMLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTISKY-
VLSAANKSNVKAAGKVGGNAPAYGAQALQRMFLSFPTTKTYFPHFDLSHGSAQQKAHQKQVANALTKAQGHLDLPGTLNLSNLHAKHLRVNPVNFKLLSHSLLVTLASHLPNTFTPAVHANLKNFLANDSTVLTISKYR

Final Score: 440
Identity: 0.7659574468085106

Results for human vs horse:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 5 1 ... -547 -551 -555]
 [-8 1 10 ... -538 -542 -546]
 ...
 [-552 -543 -534 ... 550 546 542]
 [-556 -547 -538 ... 559 555 551]
 [-560 -551 -542 ... 555 564 560]]

VLSPADKTNVKAAGKVGGAHAGEYGAEALERMLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTISKY-
VLSAADKTNVKAAGKVGGAHAGEYGAEALERMLFGPTTKTYFPHFDLSHGSAQVKAHQKVGDAITLAVGHLDLPGALSNLSDLHAHKLRVDPVNFKLLSHCLLVTLAVHLPNDFTPAVHASLDKFLSSVSTVLTISKYR

Final Score: 560
Identity: 0.8723404255319149
```

```

Results for human vs pig:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 5 1 ... -547 -551 -555]
 [-8 1 10 ... -538 -542 -546]
 ...
 [-552 -543 -534 ... 510 506 502]
 [-556 -547 -538 ... 519 515 511]
 [-560 -551 -542 ... 515 524 520]]

VLSPADKTNVKAAMGKVGAGHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKY-
VLSAADKANVKAAMGKVGAGHAGEALERMFLGFPTTKTYFPHFNLHGSDQVKAHGQKVADALTAVGHLLDLPALSAALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHHPPDFFNPSVHASLDKFLANVSTVLTSKYR

Final Score: 520
Identity: 0.8368794326241135

Results for human vs trout:

[[ 0 -4 -8 ... -564 -568 -572]
 [-4 -3 -7 ... -555 -559 -563]
 [-8 -7 -6 ... -546 -550 -554]
 ...
 [-552 -551 -542 ... 210 206 202]
 [-556 -555 -546 ... 219 215 211]
 [-560 -559 -550 ... 215 224 220]]

VLS-PA-DKTNVKAAMGK-VGAGHAGEYGAEALERMFL-SFPTTKTYFPH-FDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVH-ASLDKFLASVSTVLTSKY-
-XSLTAKDKSVVKAFAWGKISG-KADVVGAEALGRH-LTAYPQTKTYFPHFADLSPGSGPVKKGKGIIINGAIGKAVGLMDDLPGALSALSDLHAYKLRVDPVNFKLLSHCLLVTLACHHPTTEFTPAVHASLDKFFTA-VSTVLTSKYR

Final Score: 220
Identity: 0.5874125874125874

Results for human vs wolf:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 5 1 ... -547 -551 -555]
 [-8 1 10 ... -538 -542 -546]
 ...
 [-552 -543 -534 ... 497 493 489]
 [-556 -547 -538 ... 506 502 498]
 [-560 -551 -542 ... 502 511 507]]

VLSPADKTNVKAAMGKVGAGHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDK-FLASVSTVLTSKY-
VLSPADKTNIKSTWDKIGGHAGDYGGEALDRTFQSFPTTKTYFPHFDLSPGSAQVKAHGKKVADALTAVAHLLDLPALSAALSDLHAYKLRVDPVNFKLLSHCLLVTLACHHPTTEFTPAVHASLDKFFTA-VSTVLTSKYR

Final Score: 507
Identity: 0.8297872340425532

More than one sequence alignment produced the same final score and identity.

human vs cow - (score, identity): (560, 0.8723404255319149)
human vs horse - (score, identity): (560, 0.8723404255319149)

```

Questão 2-a:

Pseudocódigo

A única coisa diferente da questão anterior para essa, é a mudança no código do método `pegaValor(i, j)`, que agora não mais utiliza valores passados de `match` e `missmatch`, mas sim o valor obtido ao se alinhar os aminoácidos de acordo com a tabela `Blosum62`, obtida pela utilização da biblioteca `Bio`, no formato de dicionário. Segue o pseudocódigo desse método:


```

1 // Mesma coisa que o anterior, porém mudando o método pegaValor(i, j)
2 // que agora leva em consideração a matriz blossom62
3
4 blosum = abre o dicionário correspondente a matriz Blosum62 na lib Bio
5
6 pegaValor(i, j):
7
8     tenta:
9         valorBlosum = blosum[seq1[i], seq2[j]]
10    caso não tenha, inverte:
11        valorBlosum = blosum[seq2[j], seq1[i]]
12
13    // Ou seja: caso não tenha, por exemplo, (L, M), tenta (M, L)
14
15    valoresPossíveis = [
16        matrizAlinhamento[i-1][j-1] + valorBlosum,
17        matrizAlinhamento[i][j-1] + penalidadeGap,
18        matrizAlinhamento[i-1][j] + penalidadeGap
19    ]
20
21    retorna (max(valoresPossíveis), Operação correspondente ao max)

```

Resultados:

```

C:\Users\Pichau\Desktop\Biocomp\list 2\part 1>python e2-2a.py
Results for human vs chicken:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 1 -3 ... -548 -552 -556]
 [-8 -2 5 ... -540 -544 -548]
 ...
 [-552 -543 -535 ... 425 421 417]
 [-556 -547 -539 ... 434 430 426]
 [-560 -551 -543 ... 430 441 437]]

VLSPADKTNVKAAMGKVGAGHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSLSLHAHKL RVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKY-
MLTAEDKKLIQQAWEKAASHQEFGAEALTRMFTTYPQTKTYFPHFDLSPGSDQVRGHHGKKVLGALGNVKNVDNLSQAMAELSNLHAYNL RVDPVNFKLLSQCIQVVLAVHMGKDYTPEVHAAFDFLSAVSAVLAEKYR

Final Score: 437
Identity: 0.5886524822695035

Results for human vs cow:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 4 0 ... -548 -552 -556]
 [-8 0 8 ... -540 -544 -548]
 ...
 [-552 -544 -536 ... 618 614 610]
 [-556 -548 -540 ... 627 623 619]
 [-560 -552 -544 ... 623 634 630]]

VLSPADKTNVKAAMGKVGAGHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSLSLHAHKL RVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKY-
VLSAADKGNVKAAMGKVGGAHAEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGAKVAAALTKAVEHLDLPGALSELSDLHAHKL RVDPVNFKLLSHSLLVTLASHLP SDFTPAVHASLDKFLANVSTVLTSKYR

Final Score: 630
Identity: 0.8723404255319149

```

```

Results for human vs deer:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 4 0 ... -548 -552 -556]
 [-8 0 8 ... -540 -544 -548]
 ...
 [-552 -544 -536 ... 545 541 537]
 [-556 -548 -540 ... 554 550 546]
 [-560 -552 -544 ... 550 561 557]]

VLSPADKTNVKAAMWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSADLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKY-
VLSAANKSNVKAAMWGKVGGNAPAYGAQALQRMFLSFPTTKTYFPHFDLSHGSAQVKAHGKKVANALTKAQGHLDLPGTLSNLSNLHAHKLRVNPVNFKLLSHSLLVTLASHLPTNFTPAVHANLKNFLANDSTVLTSKYR

Final Score: 557
Identity: 0.7659574468085106

Results for human vs horse:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 4 0 ... -548 -552 -556]
 [-8 0 8 ... -540 -544 -548]
 ...
 [-552 -544 -536 ... 622 618 614]
 [-556 -548 -540 ... 631 627 623]
 [-560 -552 -544 ... 627 638 634]]

VLSPADKTNVKAAMWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSADLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKY-
VLSAADKTNVKAAMWGKVGGHAGEYGAEALERMFLGFPTTKTYFPHFDLSHGSAQVKAHGKKVGDALTAVGHLDLPGALSNDLHAHKLRVDPVNFKLLSHCLLVTLAVHLPNDFTPAVHASLDKFLSSVSTVLTSKYR

Final Score: 634
Identity: 0.8723404255319149

Results for human vs pig:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 4 0 ... -548 -552 -556]
 [-8 0 8 ... -540 -544 -548]
 ...
 [-552 -544 -536 ... 597 593 589]
 [-556 -548 -540 ... 606 602 598]
 [-560 -552 -544 ... 602 613 609]]

VLSPADKTNVKAAMWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSADLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKY-
VLSAADKANVKAAMWGKVGGAAGAHAGEALERMFLGFPTTKTYFPHFNLSHGSDQVKAHGKKVADALTAKAVGHLDLPGALSADLHAHKLRVDPVNFKLLSHCLLVTLAAHHPDDFNPSVHASLDKFLANVSTVLTSKYR

Final Score: 609
Identity: 0.8368794326241135

Results for human vs trout:

[[ 0 -4 -8 ... -564 -568 -572]
 [-4 -1 -5 ... -556 -560 -564]
 [-8 -5 -3 ... -548 -552 -556]
 ...
 [-552 -548 -540 ... 377 373 369]
 [-556 -552 -544 ... 386 382 378]
 [-560 -556 -548 ... 382 393 389]]

V-LSPADKTNVKAAMWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHF-DLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSADLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKY-
XSLTAKDKSVVKAFWKGISGKADVGAELGRMLTAYPQTKTYFSHWADLSPGSGPVKKHGGIIMGAGKAVGLMDDL VGGMSALSDLHAFKLRVDPGNFKILSHNILLVTLAIHFPSDFTEPVHIAVDKFLAAVSAALADKYR

Final Score: 389
Identity: 0.5594405594405595

Results for human vs wolf:

[[ 0 -4 -8 ... -556 -560 -564]
 [-4 4 0 ... -548 -552 -556]
 [-8 0 8 ... -540 -544 -548]
 ...
 [-552 -544 -536 ... 599 595 591]
 [-556 -548 -540 ... 608 604 600]
 [-560 -552 -544 ... 604 615 611]]

VLSPADKTNVKAAMWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSADLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKY-
VLSPADKTNIKSTWDKIGGHAGDYGGEALDRTFQSFPPTTKTYFPHFDLSPGSAQVKAHGKKVADALTAVAHLDLPGALSADLHAYKLRVDPVNFKLLSHCLLVTLACHHPTEFTPAVHASLDKFFTA VSTVLTSKYR

Final Score: 611
Identity: 0.8226950354609929

The best score was achieved when human vs horse were compared.
Obtained score: 634
Obtained identity: 0.8723404255319149

```