

BT3040 – BIOINFORMATICS – Assignment 7

Submitted by Sahana (BE17B038)

Question 1

Algorithm –

- Based on hydrophobicity values (which are stored in a dictionary), a list containing the Hgm values of all residues is created.
- With AA residue on x-axis and Hgm values on y-axis, the Hydrophobicity profile is plotted.
- Depending on whether the Hgm value for a particular residue is higher/lower than the mean Hgm value, a binary list is created where 1 - Hydrophobic. 0 - Hydrophilic.
- This binary list will help us identify Alpha and Beta strands in the given sequence.
- An alpha strand is identified as a repeating unit of 2 values of Hgm above the mean and 2 values below the mean (axis).
- A beta strand is identified as alternative values of Hgm above and below the mean axis.

Code –

```
import matplotlib.pyplot as plt
import numpy as np

def hydrophobicity_profile(seq):
    Hgm = {'A': 13.85, 'D': 11.61, 'C': 15.37, 'E': 11.38, 'F': 13.93,
           'G': 13.34, 'H': 13.82, 'I': 15.28, 'K': 11.58, 'L': 14.13,
           'M': 13.86, 'N': 13.02, 'P': 12.35, 'Q': 12.61, 'R': 13.10,
           'S': 13.39, 'T': 12.70, 'V': 14.56, 'W': 15.48, 'Y': 13.88}
    H_val = []

    n = len(seq)
    for i in range(n):
        H_val.append(Hgm[seq[i]])

    plt.plot(H_val, '-o')
    plt.xlabel('AA residue')
    plt.ylabel('Hydrophobicity values')
    plt.title('Hydrophobicity profile')
    plt.plot(np.mean(H_val)*np.ones(n), 'r')
    plt.show

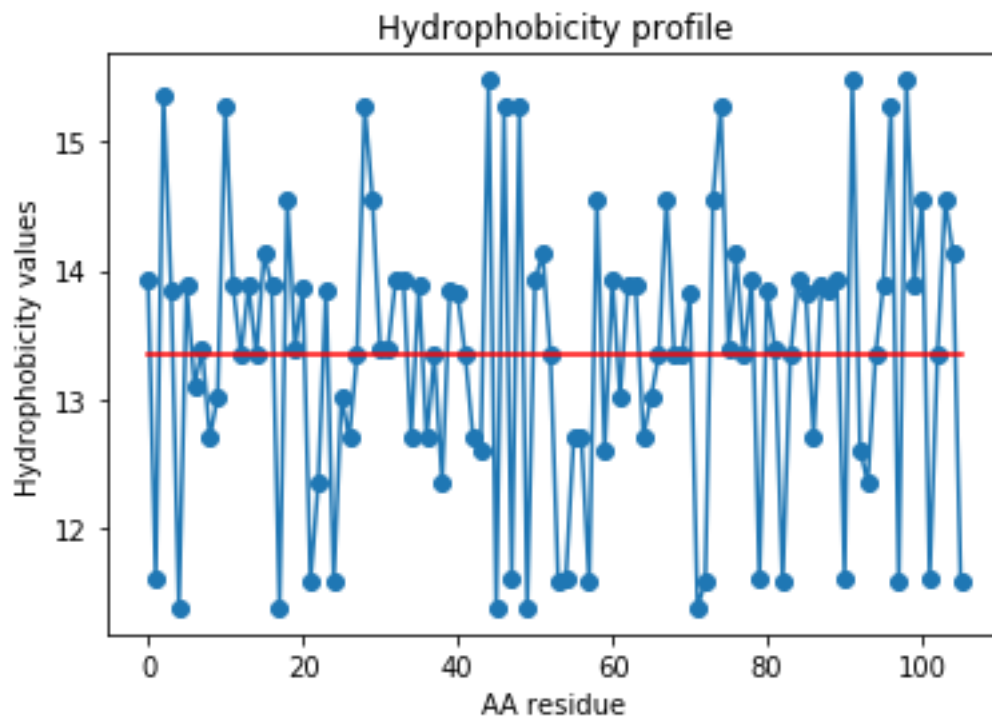
    print('The pattern of hydrophobicity values are printed. 1 -
Hydrophobic. 0 - Hydrophilic.\n')
    binary = [0]*n
    for i in range(n):
        if H_val[i]>np.mean(H_val):
            binary[i]=1
    print(binary)
    return binary

a =
'FDCAEYRSTNIYGYGLYEVSMPKPAKNTGIVSSFFTYTGPAGHTQWEIDIEFLGKDTTKVQFNYYTNGVGGHEKV
ISLGFDASKGFHTYAFDWQPGYIKWYVDGVLK'
hydrophobicity_profile(a)
```

Output –

The pattern of hydrophobicity values are printed. 1 - Hydrophobic. 0 - Hydrophilic.

```
[1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 0]
```



Alpha strands –

- Position 9 – 0, 0, 1, 1 – TNIY
- Position 20 – 1, 1, 0, 0 – SKMP
- Position 27 – 0, 0, 1, 1 – TGIV
- Position 38 – 0, 0, 1, 1, 0, 0 – GPAHGT
- Position 51 – 1, 1, 0, 0 – FLGK
- Position 63 – 1, 1, 0, 0 – YYTN
- Position 72 – 0, 0, 1, 1 – EKVI
- Position 81 – 1, 1, 0, 0, 1, 1 – ASKGFH
- Position 94 – 0, 0, 1, 1 – PGYI
- Position 100 – 1, 1, 0, 0, 1, 1 – YVDGVLK

Beta strands – (4 residues or more is only considered)

- Position 4 – 1, 0, 1, 0, 1, 0 – AEYRST
- Position 12 – 1, 0, 1, 0, 1 – YGYGL
- Position 34 – 1, 0, 1, 0 – FTYT
- Position 44 – 0, 1, 0, 1, 0, 1, 0, 1 – QWEIDIEF
- Position 58 – 0, 1, 0, 1, 0, 1 – KVQFNY
- Position 77 – 1, 0, 1, 0, 1 – LGFDA

- Position 90 - 1, 0, 1, 0 - FDWQ

Question 2 –

Algorithm –

- From the previous function, we obtain binary values and Hydrophobicity values.
- We note that there are no alpha strands of length more than 8 residues.
- Hence, amphipathicity of beta strands is calculated as follows. It is the absolute difference between average hydrophobicity values of residues whose values are above the mean value and the average hydrophobicity values of residues whose values are below the mean value.

Code –

```
def amphipathicity(a):
    [binary,H_val] = hydrophobicity_profile(a) #Binary format
    n=len(a)

    beta = []
    alpha = []
    for i in range(n):
        if binary[i:i+6]==[0,1,0,1,0,1] or binary[i:i+6]==[1,0,1,0,1,0]:
            beta.append(i)
        if binary[i:i+8]==[0,0,1,1,0,0,1,1] or
binary[i:i+8]==[1,1,0,0,1,1,0,0]:
            alpha.append(i)
    amph_beta = [0]*len(beta)
    for i in range(0,len(beta)):
        b1=0
        b2=0
        start = binary[beta[i]]
        for j in range(0,6,2):
            b1+= H_val[beta[i]+j]
            b2+= H_val[beta[i]+j+1]
        b1=b1/3
        b2=b2/3
        if start ==1:
            amph_beta[i] = b1-b2
        else:
            amph_beta[i] = b2-b1

    if alpha==[]:
        print('There are no alpha helices of stretch 8 residues.\n')
    if beta==[]:
        print('There are no beta strands')
    else:
        print('Beta strands of length 6 is found at positions:')
        print(beta)
        print('The corresponding amphipathicity values are:')
        print(amph_beta)

a =
'FDCAEYRSTNIYGYGLYEVSMPAKNTGIVSSFFTYTGPAHGTQWEIDIEFLGKDTTKVQFNYYTNGVGGHEKV
ISLGFDASKGFHTYAFDWQPGYIKWYVDGVLK'
amphipathicity(a)
```

Output –

There are no alpha helices of stretch 8 residues.

Beta strands of length 6 is found at positions:

[3, 43, 44, 45, 57]

The corresponding amphipathicity values are:

[1.313333333333361, 3.479999999999986, 3.889999999999997, 3.373333333333295, 1.7200000000000042]

Calculations -

Position 3 - $H_val = [13.85, 11.38, 13.88, 13.1, 13.39, 12.7]$

$b1 = (13.85+13.38+13.39)/3 = 13.706666666666669$

$b2 = (11.38+13.1+12.7)/3 = 12.393333333333333$

amphipathicity of position 3 = $|b1-b2| = 1.313333333333361$

Position 43 - $H_val = [12.61, 15.48, 11.38, 15.28, 11.61, 15.28]$

$b1 = (12.61+11.38+11.61)/3 = 11.866666666666667$

$b2 = (15.48+15.28+15.28)/3 = 15.346666666666666$

amphipathicity of position 43 = $|b1-b2| = 3.479999999999986$

Position 44 - $H_val = [15.48, 11.38, 15.28, 11.61, 15.28, 11.38]$

$b1 = (15.48+15.28+15.28)/3 = 15.346666666666666$

$b2 = (11.38+11.61+11.38)/3 = 11.456666666666669$

amphipathicity of position 44 = $|b1-b2| = 3.8899999999999997$

Position 45 - $H_val = [11.38, 15.28, 11.61, 15.28, 11.38, 13.93]$

$b1 = (11.38+11.61+11.38)/3 = 11.456666666666669$

$b2 = (15.28+15.28+13.93)/3 = 14.829999999999998$

amphipathicity of position 45 = $|b1-b2| = 3.373333333333295$

Position 57 - $H_val = [11.58, 14.56, 12.61, 13.93, 13.02, 13.88]$

$b1 = (11.58+12.61+13.02)/3 = 12.403333333333333$

$b2 = (14.56+13.93+13.88)/3 = 14.123333333333335$

amphipathicity of position 57 = $|b1-b2| = 1.7200000000000042$

Question 3 -

Algorithm -

- For a specified window length d, Hydrophobicity value of a residue is taken as average of $H_val(s)$ from $i-(d/2)$ to $i+(d/2)$
- The H_val for remaining residues are from the dictionary itself.
- The Hydrophobicity profile is then plotted.

Code -

```
def hgm_window(seq,d):  
    Hgm = {'A': 13.85, 'D': 11.61, 'C': 15.37, 'E': 11.38, 'F': 13.93,  
           'G': 13.34, 'H': 13.82, 'I': 15.28, 'K': 11.58, 'L': 14.13,
```

```

        'M': 13.86, 'N': 13.02, 'P': 12.35, 'Q': 12.61, 'R': 13.10,
        'S': 13.39, 'T': 12.70, 'V': 14.56, 'W': 15.48, 'Y': 13.88}
H_val = []
l = int(d/2)
n = len(seq)
for i in range(l):
    H_val.append(Hgm[seq[i]])
for i in range(l,n-l-1):
    a=0
    for j in range(-l,l,1):
        a+=Hgm[seq[i+j]]
    a=a/d
    H_val.append(a)
for i in range(n-l-1,n):
    H_val.append(Hgm[seq[i]])

plt.plot(H_val, '-o')
plt.xlabel('AA residue')
plt.ylabel('Hydrophobicity values')
plt.title('Hydrophobicity profile with window length')
plt.plot(np.mean(H_val)*np.ones(n), 'r')
plt.show

```

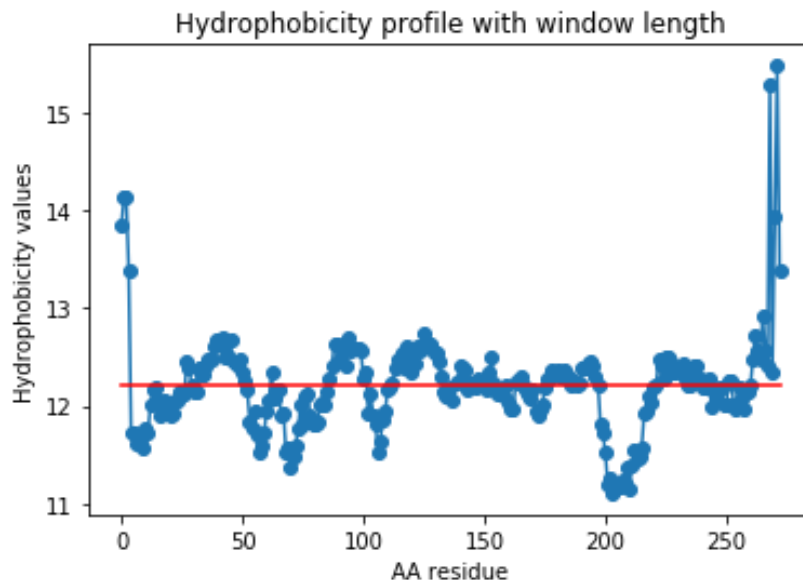
```

a =
'ALLSFERKYRVRGGTLIGGDLFDWFVGPYFVGFFGVSAIFFIFLGVSLIGYAASQGPTWDPFAISINPPDLKYG
LGAAPLLEGGFWQAITVCALGAFISWMLREVEISRKLIGIGWHVPLAFCVPPIFMFCVLQVFRPLLLGSWGHAFFPYG
ILSHLDWVNNFGYQYLNWHYNPGHMSSVSFLFVNAMALGLHGGLILSVANPGDGDVKVKTAEHENQYFRDVGYSI
GALSIHRLGLFLASNIFLTGAFGTIASGPFWTRGWPEWWGWLLDIPFWS'
hgm_window(a,9)
hgm_window(a,19)

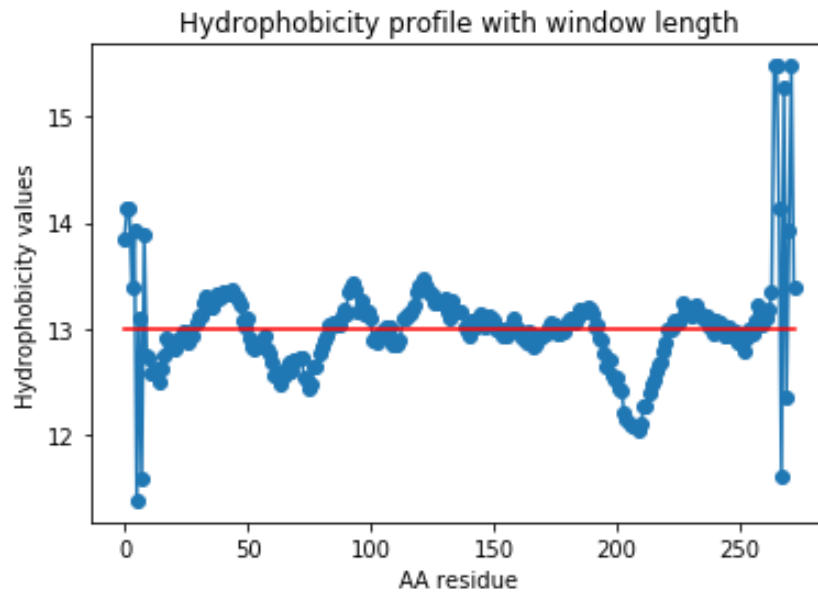
```

Output -

Window length = 9



Window length = 19



Transmembrane segments -

Since both the profiles (with window lengths 9 and 19) display highly hydrophobic residues for interior and polar residues for the exterior, they are membrane proteins, since there's a mixture of alpha and beta strands.

Question 4 -

(a) Pattern – S-T-V-[IL](2)-{DERK}

Result - 138 hits in 138 sequences.

Approximate number of expected random matches in ~ 100'000 sequences (50'000'000 residues): 212

1NI5-A	3KCU-E	3O91-D	3Q7K-K	4RIB-A	5MLQ-B
1Z2M-A	3KCV-A	3O92-A	3R66-C	4RIB-B	5TJ5-L
2AL3-A	3KCV-B	3O92-B	3R66-D	4RIC-A	5VOX-c
2P68-A	3KCV-C	3O92-C	3RT3-B	4RIC-B	5VOY-c
2P68-B	3KCV-D	3O92-D	3SDL-C	4RID-A	5VOZ-c
2PNF-A	3KCV-E	3O93-A	3SDL-D	4RID-B	5WOX-A
2PNF-B	3KCV-F	3O93-B	3U7S-A	4RY3-A	6BI8-C
2R9Y-A	3KCV-G	3O93-C	3U7S-B	4U7U-A	6BI8-D
2ZZV-A	3KCV-H	3O93-D	4C0T-A	4U7U-M	6C6L-M
2ZZV-B	3KCV-I	3O94-A	4DRS-A	4XRI-A	6EOJ-A
2ZZW-A	3KCV-J	3O94-B	4DRS-B	4XRK-A	6G7D-A
2ZZW-B	3MA8-A	3O94-C	4HUK-A	5B18-A	6G80-A
2ZZX-A	3MA8-B	3O94-D	4HUL-A	5B18-B	6G80-B
2ZZX-B	3ML0-B	3PE7-A	4HUM-A	5B18-C	6G80-C
2ZZX-C	3NKL-A	3PSE-B	4HUN-A	5B18-D	6G80-M
2ZZX-D	3NKL-B	3Q7K-A	4MHL-A	5C6P-A	6H40-A
3AYF-A	3O90-A	3Q7K-B	4QYZ-A	5CD4-I	6O7T-e
3AYG-A	3O90-B	3Q7K-C	4RI8-A	5CD4-U	6O7U-e

3K3W-B	3O90-C	3Q7K-D	4RI8-B	5H9E-A	6O7V-e
3KCU-A	3O90-D	3Q7K-E	4RI9-A	5H9F-A	6O7W-e
3KCU-B	3O91-A	3Q7K-H	4RI9-B	5J44-A	6O7X-e
3KCU-C	3O91-B	3Q7K-I	4RIA-A	5J44-B	6P0Y-A
3KCU-D	3O91-C	3Q7K-J	4RIA-B	5MLQ-A	6P0Y-B

(b) Pattern - [FILV]-Q-x-x-x-[RK]-G-x-x-x-[RK]-x-x-[FILVWY]

Result – 219 hits in 214 sequences.

Approximate number of expected random matches in ~ 100'000 sequences
(50'000'000 residues): 125

1B7T-A	1SCM-A	2IXE-A	3I5F-A	3MIL-B	4BYF-C	4R8G-E	5F0P-A	5TBY-B
1BR1-A	1SR6-A	2IXE-D	3I5G-A	3N0U-A	4E50-A	4RU1-A	5JPM-A	5V7X-A
1BR1-C	1T9H-A	2IXF-A	3I5H-A	3N0U-B	4E53-A	4RU1-B	5JPM-D	5X6O-G
1BR1-E	1TLV-A	2IXF-B	3I5I-A	3N0U-C	4E53-B	4RU1-C	5JPN-A	6C1D-P
1BR1-G	1VRB-A	2IXF-C	3IBX-A	3PN7-A	4FXG-A	4RU1-D	5JTW-A	6C1G-P
1BR4-A	1VRB-B	2IXF-D	3IBX-D	3PN7-D	4FXG-D	4RU1-E	5JTW-D	6C1H-P
1BR4-C	1VRB-C	2IXG-A	3J04-A	3TS5-A	4FXK-A	4RU1-F	5LPU-A	6EUD-A
1BR4-E	1VRB-D	2OS8-A	3J04-D	3TS5-D	4HRE-A	4RU1-G	5LPU-B	6FSA-A
1BR4-G	1W7B-A	2OTG-A	3JAX-A	3TUY-A	4HRE-B	4RU1-H	5LPX-A	6H7W-Q
1DFK-A	1W7I-A	2RD3-A	3JAX-B	3TUY-D	4HRE-C	4RU1-I	5LQ0-A	6H7W-R
1DFL-A	1W7J-A	2RD3-D	3JTD-A	3WFN-B	4HRE-D	4RU1-J	5LQ0-B	6HEG-A
1DFL-B	1WDC-A	2W4T-C	3JVT-A	3WFN-C	4K8O-A	4RU1-K	5LQ2-A	6J7C-A
1H99-A	1XJL-A	2W4V-C	3KZW-A	3WFN-D	4L79-A	4RU1-L	5LQ2-B	6JXA-A
1I84-S	1XJL-B	2W4W-C	3KZW-B	3WFN-E	4LZX-B	4X9P-A	5M6B-A	6JXC-A
1I84-V	2BL0-A	2WFT-A	3KZW-C	3WQC-A	4N18-A	4XAM-A	5M6B-B	
1KK7-A	2DDI-A	2WFT-B	3KZW-D	3WQC-B	4NN2-A	4XAM-B	5M6B-D	
1KK8-A	2DDJ-A	2WG3-C	3KZW-E	3WQD-A	4NN2-B	4ZLK-A	5N69-A	
1KQM-A	2DFS-A	2WG3-D	3KZW-F	3WQD-B	4OUA-A	5F0J-A	5N69-B	
1KWO-A	2DFS-M	3DTP-A	3KZW-G	3WQE-A	4OUA-B	5F0K-A	5N7D-A	
1L2O-A	2EC6-A	3DTP-B	3KZW-H	3WQE-B	4PB3-A	5F0K-B	5N7D-B	
1M45-B	2HYU-A	3HO3-A	3KZW-I	3WQF-A	4PB3-B	5F0K-C	5N7F-A	
1N2D-C	2HYV-A	3HO4-A	3KZW-J	3WQF-B	4PB4-A	5F0K-D	5N7F-B	
1OE9-A	2HYW-A	3HO4-B	3KZW-K	3WQG-A	4PB4-B	5F0K-E	5N7G-A	
1QVI-A	2HYW-B	3HO5-A	3KZW-L	3WQG-B	4PB5-A	5F0L-A	5N7G-B	
1S5G-A	2IX7-C	3HO5-B	3MIL-A	4BYF-A	4PB5-B	5F0M-A	5TBY-A	

Question 5 –

Algorithm –

- The FASTA file is read.
- The contents of the file is then split into 2 lists, one containing the headers and the other containing the sequences.
- The occurrences of the patterns from Question 4 are checked through all sequences from the above list.
- If found, the position and the found pattern is printed. Else, 'Not found!' is printed.

Code –

```
def pattern_search():
    f = open('D:\Textbook-Sem6\Bioinformatics\Q5.fasta','r') #While testing
    this code, please give the relavent address.
    a = ''
    sequences = []
    header = []
    for line in f:
        if line[0]=='>':
            header.append(line)
            sequences.append(a)
            a=''
        else:
            a+=str(line)
    sequences.append(a)
    sequences.pop(0)
    for i in range(len(sequences)):
        sequences[i]= sequences[i].replace('\n','')
    for j in range(len(header)):
        header[j]= header[j].replace('\n','')
    n = len(header)

    a = ['I','L']
    b = ['D','E','R','K']
    c = ['F','I','L','V']
    d = ['R','K']
    e = ['F','I','L','V','W','Y']

    p1_len = 6
    p2_len = 14

    print('Searching for the pattern - "S-T-V-[IL](2)-{DERK}" in the given
sequence.\nPrinting position and the pattern in the sequence.')
    for i in range(n):
        p1 = []
        print(header[i])
        s = sequences[i]
        for i in range(len(s)-p1_len+1):
            if s[i:i+3]=='STV' and s[i+3] in a and s[i+4] in a and s[i+5]
not in b:
                print(i,s[i:i+p1_len])
                p1.append(s[i:i+p1_len])
            if len(p1)==0:
                print('Not found!')

    print('\nSearching for the pattern - "[FILV]-Q-x-x-x-[RK]-G-x-x-x-[RK]-
x-x-[FILVWY]" in the given sequence.\nPrinting position and the pattern in
the sequence.')
    for i in range(n):
        p2 = []
        print(header[i])
        s = sequences[i]
        for i in range(len(s)-p2_len+1):
            if s[i] in c and s[i+1]=='Q' and s[i+5] in d and s[i+6]=='G'
and s[i+10] in d and s[i+13] in e:
                print(i,s[i:i+p2_len])
                p2.append(s[i:i+p2_len])
            if len(p2)==0:
                print('Not found!')
```



```
f.close()
```

```
pattern_search() #Running the function
```

Output -

```
Searching for the pattern - "S-T-V-[IL](2)-{DERK}" in the given
sequence.
Printing position and the pattern in the sequence.
>1BR1:A|PDBID|CHAIN|SEQUENCE

Not found!
>1B7T:A|PDBID|CHAIN|SEQUENCE

Not found!
>1BR4:A|PDBID|CHAIN|SEQUENCE

Not found!
>1H99:A|PDBID|CHAIN|SEQUENCE

Not found!
>1DFL:B|PDBID|CHAIN|SEQUENCE

Not found!
>1I84:S|PDBID|CHAIN|SEQUENCE

Not found!
>1NI5:A|PDBID|CHAIN|SEQUENCE

25 STVLLH
>2P6B:B|PDBID|CHAIN|SEQUENCE

Not found!
>2PNF:B|PDBID|CHAIN|SEQUENCE

31 STVIIT
>2ZZB:B|PDBID|CHAIN|SEQUENCE
Not found!

Searching for the pattern - "[FILV]-Q-x-x-x-[RK]-G-x-x-x-[RK]-x-
x-[FILVWY]" in the given sequence.
Printing position and the pattern in the sequence.
>1BR1:A|PDBID|CHAIN|SEQUENCE

797 FQAQCRGYLARKAF
>1B7T:A|PDBID|CHAIN|SEQUENCE

784 FQAHIRGYLIRKAY
>1BR4:A|PDBID|CHAIN|SEQUENCE

797 FQAQCRGYLARKAF
>1H99:A|PDBID|CHAIN|SEQUENCE

51 IQRNQKGLDIKNAL
>1DFL:B|PDBID|CHAIN|SEQUENCE

780 FQAHIRGYLIRKAY
>1I84:S|PDBID|CHAIN|SEQUENCE
```

```

797 FQAQCRGYLARKAF
>1NI5:A|PDBID|CHAIN|SEQUENCE

Not found!
>2P6B:B|PDBID|CHAIN|SEQUENCE

Not found!
>2PNF:B|PDBID|CHAIN|SEQUENCE

Not found!
>2ZZB:B|PDBID|CHAIN|SEQUENCE

Not found!

```

Question 6 –

Algorithm –

- Download the protein sequences (in FASTA format) of the given UNIPROT IDs from Retrieve ID Mapping at <https://www.uniprot.org/uploadlists/>
- The search pattern is slight modified as [KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW] to suit the first web browser. From the second one, search motif is [KRHQNST]-x-G-[ILVFMYWAC]-x-[IVLFMYW]-x-[IVLFMYW]
- Paste the FASTA file in the given space and submit your input. The server will output the matches as specified below.

Output –

Summary of results from both the servers for question 6

UNIPROT ID	Server 1	Server 2
P66948	0	0
P0A910	1	1
P06996	1	1
P0A940	2	2
P0A937	0	0
P9WIU5	1	1
P9WHT9	0	0
P03304	5	5
Q66765	5	5
P12296	3	3
Q155Z9	2	2
P17594	5	5

Results from - https://www.bioinformatics.org/sms2/protein_pattern.html

Results for 487 residue sequence "sp|P66948|BEPA_ECOLI Beta-barrel assembly-enhancing protease OS=Escherichia coli (strain K12) OX=83333 GN=bepA PE=1 SV=1" starting "MFRQLKKNLV"

no matches found for this sequence.

Results for 346 residue sequence "sp|P0A910|OMPA_ECOLI Outer membrane protein A OS=Escherichia coli (strain K12) OX=83333 GN=ompA PE=1 SV=1" starting "MKKTAIAIAV"

>match number 1 to "[KRHQNST].G[ILVFMWAC].[IVLFMYW].[IVLFMYW]" start=184
end=191
SLGVSYRF

Results for 367 residue sequence "sp|P06996|OMPC_ECOLI Outer membrane porin C OS=Escherichia coli (strain K12) OX=83333 GN=ompC PE=1 SV=1" starting "MKVKVLSLLV"

>match number 1 to "[KRHQNST].G[ILVFMWAC].[IVLFMYW].[IVLFMYW]" start=114
end=121
NYGVVYDV

Results for 810 residue sequence "sp|P0A940|BAMA_ECOLI Outer membrane protein assembly factor BamA OS=Escherichia coli (strain K12) OX=83333 GN=bamA PE=1 SV=1" starting "MAMKKLLIAS"

>match number 1 to "[KRHQNST].G[ILVFMWAC].[IVLFMYW].[IVLFMYW]" start=251
end=258
KKGIVYTV

>match number 2 to "[KRHQNST].G[ILVFMWAC].[IVLFMYW].[IVLFMYW]" start=769
end=776
SAGIALQW

Results for 113 residue sequence "sp|P0A937|BAME_ECOLI Outer membrane protein assembly factor BamE OS=Escherichia coli (strain K12) OX=83333 GN=bamE PE=1 SV=1" starting "MRCKTLTAAA"

no matches found for this sequence.

Results for 326 residue sequence "sp|P9WIU5|ARFA_MYCTU Peptidoglycan-binding protein ArfA OS=Mycobacterium tuberculosis (strain ATCC 25618 / H37Rv) OX=83332 GN=arfA PE=1 SV=1" starting "MASKAGLGQT"

>match number 1 to "[KRHQNST].G[ILVFMWAC].[IVLFMYW].[IVLFMYW]" start=266
end=273
SEGINIPL

Results for 291 residue sequence "sp|P9WHT9|PSB_MYCTU Proteasome subunit beta OS=Mycobacterium tuberculosis (strain ATCC 25618 / H37Rv) OX=83332 GN=prcB PE=1 SV=1" starting "MTWPLPDRLS"

no matches found for this sequence.

Results for 2290 residue sequence "sp|P03304|POLG_EMCV Genome polyprotein OS=Encephalomyocarditis virus OX=12104 PE=1 SV=1" starting "MATTMEQETC"

>match number 1 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=242
end=249
KTGWRVQV

>match number 2 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=760
end=767
NHGLLVRW

>match number 3 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=921
end=928
HRGLEVRL

>match number 4 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=1756
end=1763
KAGVSVPV

>match number 5 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=2141
end=2148
RAGLYLTY

**Results for 2292 residue sequence "sp|Q66765|POLG_EMCVR Genome polyprotein
OS=Encephalomyocarditis virus (strain Rueckert) OX=650129 PE=1 SV=1"
starting "MATTMEQETC"**

>match number 1 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=244
end=251
KTGWRVQV

>match number 2 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=762
end=769
NHGLLVRW

>match number 3 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=923
end=930
HRGLEVRL

>match number 4 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=1758
end=1765
KAGVSVPV

>match number 5 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=2143
end=2150
RAGLYLTY

**Results for 2293 residue sequence "sp|P12296|POLG_ENMGO Genome polyprotein
OS=Mengo encephalomyocarditis virus OX=12107 PE=1 SV=3" starting
"MATTMEQEIC"**

>match number 1 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=244
end=251
KTGWRVQV

>match number 2 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=1759
end=1766
KAGVSVPV

>match number 3 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=2144
end=2151
RAGLYLTY

Results for 2181 residue sequence "sp|Q155Z9|POLG_SVV1 Genome polyprotein OS=Seneca Valley virus (isolate -/United States/SSV-001/2002) OX=686944 PE=1 SV=1" starting "MQNSHFSFDT"

>match number 1 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=69
end=76
QLGLDIVY

>match number 2 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=262
end=269
KCGWQVQV

Results for 2292 residue sequence "sp|P17594|POLG_EMCDV Genome polyprotein OS=Encephalomyocarditis virus (strain emc-d diabetogenic) OX=12106 PE=3 SV=2" starting "MATTMEQEIC"

>match number 1 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=244
end=251
KTGWRVQV

>match number 2 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=762
end=769
NHGLLVRW

>match number 3 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=923
end=930
HGGLEIRL

>match number 4 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=1758
end=1765
KAGVSPV

>match number 5 to "[KRHQNST].G[ILVFMYWAC].[IVLFMYW].[IVLFMYW]" start=2143
end=2150
RAGLYLTY

Result from <https://prosite.expasy.org/scanprosite/>

Total number of matches: 25 hits in 9 sequences
They are -

sp-P0A910-OMPA_ECOLI	184	191	USERPAT1	.	.	.	SlGVsYrF
sp-P06996-OMPC_ECOLI	114	121	USERPAT1	.	.	.	NyGVvYdV
sp-P0A940-BAMA_ECOLI	251	258	USERPAT1	.	.	.	KkGIyVtV
sp-P0A940-BAMA_ECOLI	769	776	USERPAT1	.	.	.	SaGIaLqW
sp-P9WIU5-ARFA_MYCTU	266	273	USERPAT1	.	.	.	SeGInIpL
sp-P03304-POLG_EMCV	242	249	USERPAT1	.	.	.	KtGWvVqV
sp-P03304-POLG_EMCV	760	767	USERPAT1	.	.	.	NhGLlVrW
sp-P03304-POLG_EMCV	921	928	USERPAT1	.	.	.	HrGLeVrL
sp-P03304-POLG_EMCV	1756	1763	USERPAT1	.	.	.	KaGVsVpV
sp-P03304-POLG_EMCV	2141	2148	USERPAT1	.	.	.	RaGLyLtY
sp-Q66765-POLG_EMCVR	244	251	USERPAT1	.	.	.	KtGWvVqV
sp-Q66765-POLG_EMCVR	762	769	USERPAT1	.	.	.	NhGLlVrW

sp-Q66765-POLG_EMCV	923	930	USERPAT1	.	.	.	HrGLeVrL
sp-Q66765-POLG_EMCV	1758	1765	USERPAT1	.	.	.	KaGVsVpV
sp-Q66765-POLG_EMCV	2143	2150	USERPAT1	.	.	.	RaGLyLtY
sp-P12296-POLG_ENMGO	244	251	USERPAT1	.	.	.	KtGWrVqV
sp-P12296-POLG_ENMGO	1759	1766	USERPAT1	.	.	.	KaGVsVpV
sp-P12296-POLG_ENMGO	2144	2151	USERPAT1	.	.	.	RaGLyLtY
sp-Q155Z9-POLG_SVV1	69	76	USERPAT1	.	.	.	QlGLdIvY
sp-Q155Z9-POLG_SVV1	262	269	USERPAT1	.	.	.	KcGWqVqV
sp-P17594-POLG_EMCD	244	251	USERPAT1	.	.	.	KtGWrVqV
sp-P17594-POLG_EMCD	762	769	USERPAT1	.	.	.	NhGLlVrW
sp-P17594-POLG_EMCD	923	930	USERPAT1	.	.	.	HgGLeIrL
sp-P17594-POLG_EMCD	1758	1765	USERPAT1	.	.	.	KaGVsVpV
sp-P17594-POLG_EMCD	2143	2150	USERPAT1	.	.	.	RaGLyLtY