**SET - A**

**Name: ID:**

**Q1.** Write a Python program that takes o**ne number as input** from the user, this number indicates the number of string/strings to be taken as input. Then you have to find the **character with the highest number of occurrences combining all the strings** taken as input. \*\*(If you find **more than 1 characters** which shares the highest number of occurrences in this case consider the **first character** to be the target character.)\*\* Then check if the occurrence of that particular character is **even or odd**. If **odd** then you should print the **combining string from the first step as an output where all the occurrence of that common character is gone**. If the value is even then the output shall be the combined string in **reverse order and all the occurrence of that common character is gone.**

**\*\*You can’t use the built-in reverse() function or list\*\***

| **sample input** | **sample output** |
| --- | --- |
| **3**  **dog**  **cat**  **lion** | **o occurred 2 times**  **niltacgd** |
| **2**  **tiger**  **deer** | **e occurred 3 times**  **tigrdr** |

**Explanation**: The character “o” has the highest number of occurrences in the combined string “dogcatlion”.

**Explanation**: In the combined string “tigerdeer”, character “e” has the highest number of occurrences.

**SET - A**

**Name: ID:**

| | **1** | **i = 10** | | --- | --- | | **2** | **sum = 0** | | **3** | **p = 2** | | **4** | **while (i>0):** | | **5** | **print('<---', end='')** | | **6** | **j = 0** | | **7** | **s = ''** | | **8** | **while(j<70) :** | | **9** | **sum += 7//2\*i+(p+3)%5+i%3** | | **10** | **if j%4 ==0:** | | **11** | **s += ':)'** | | **12** | **else:** | | **13** | **s += ':('** | | **14** | **i-2** | | **15** | **print(sum, end='')** | | **16** | **j += sum//3** | | **17** | **p += 1** | | **18** | **print('--->')** | | **19** | **print(s)** | | **20** | **i -= 2** | | | **Outputs** | | --- | |  | |  | |  | |  | |  | |  | |  | |  | |  | |  | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**SET - B**

**Name: ID:**

**Q1. String Builder v2:**

You will be given one string as an input which is the result of the multiplication string of a certain string. Ie. **“wowowowowo”** is basically **“wo”\*5.** You have to find out the **core substring** and **how many times it was multiplied by to get to the input string** and then make a **pattern** like the output which will take the number of multiplication as the **row numbers**.

**\*\*it’s fixed that the core substring will not have any duplicate values\*\***

| **input** | **output** |
| --- | --- |
| **“abcdabcdabcdabcd”** | **The given string was a multiplied form of : abcd \*4**  **a**  **ab**  **abc**  **abcd** |
| **“CSE321CSE321CSE321”** | **The given string was a multiplied form of : CSE321 \*3**  **C**  **CS**  **CSE** |
| **“xyzwxyzwxyzwxyzwxyzwxyzw”** | **The given string was a multiplied form of : xyzw \*6**  **x**  **xy**  **xyz**  **xyzw**  **Can't print more lines as the string ran out.** |

**SET - B**

**Name: ID:**

| | **1** | **i = 5** | | --- | --- | | **2** | **sum = 10** | | **3** | **p = 8** | | **4** | **while (i>0):** | | **5** | **print('<---', end='')** | | **6** | **j = 10** | | **7** | **s = ''** | | **8** | **while(j<60):** | | **9** | **sum += 9//4\*i+(p+3)%5+i%2** | | **10** | **if j%5 ==0:** | | **11** | **s += ':)'** | | **12** | **else:** | | **13** | **s += ':('** | | **14** | **i-1** | | **15** | **print(sum, end='')** | | **16** | **j += sum//2** | | **17** | **p += 2** | | **18** | **print('--->')** | | **19** | **print(s)** | | **20** | **i -= 1** | | | **Outputs** | | --- | |  | |  | |  | |  | |  | |  | |  | |  | |  | |  | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |