Assignment 3

October 30, 2024

[22]:
```
# Write a program that takes the user's name and pan card number as input.␣
 ↪Validate the information using isX function and print the details.

# This function checks if length of pan card number is 10
def check_pan(pan_num):
    l = 1
    while(int(pan_num/10) > 0):
        l += 1
        pan_num = int(pan_num/10)
    print(l)
    return l

def isX(name, pan_num):
    if check_pan(pan_num) != 10:
        print("Invalid Pan Number")
        return
    print("\n----------Details of User----------\n")
    print("Name of the user: ", name)
    print("Pan Card number: ", pan_num)

name = input("Enter the user's name: ")
pan_num = int(input("Enter the pan card number: "))

isX(name, pan_num)
```

```
Enter the user's name: John Doe
Enter the pan card number: 1234567890
10

----------Details of User----------

Name of the user:  John Doe
Pan Card number:  1234567890
```

[28]:
```
# Write a program to generate an Abecedarian series. (a series in which␣
 ↪elements appears in an alphabetical order)

def is_abecedarian(word):
```

```
    return list(word) == sorted(word)

def generate_abecedarian_series(words):
    l = []
    for i in words:
        if is_abecedarian(i):
            l.append(i)
    return l

words = ["cbad", "loop", "almost", "bag", "ace", "cry", "cancer", "biopsy"]
abecedarian_words = generate_abecedarian_series(words)
print("Abecedarian series:", abecedarian_words)
```

```
Abecedarian series: ['loop', 'almost', 'ace', 'cry', 'biopsy']
```

[3]:
```
# Write a program that counts the occurrences of a character in a string. Do␣
 ↪not use built in functions.

str1 = input("Enter the string: ")
ch = input("Enter the character to be searched and counted in the string: ")
c = 0 # Stores the count of given character in the string
for i in str1:
    if i == ch:
        c +=1
print(f'No of occurrences of "{ch}" in the given string "{str1}" is: {c}')
```

```
Enter the string: Adam is a good boy
Enter the character to be searched and counted in the string: o
No of occurrences of "o" in the given string "Adam is a good boy" is: 3
```

[6]:
```
# Write a function that takes a list of words and returns the length of the␣
 ↪longest one.
str1 = input("Enter the string: ")
ls = str1.split(" ")
maxlen = 0
for i in ls:
    s = len(i)
    if s > maxlen:
        maxlen = s
        word = i
print(f'Word with longest length in input string is: {word}')
```

```
Enter the string: The judge sentenced her to three months in prison for
shoplifting
Word with longest length in input string is: shoplifting
```

[11]:
```
# Write a function to get the first half of half of a specified string of even␣
 ↪length.
```

```
str1 = "The judge sentenced her to four months in prison for shoplifting"
a = len(str1)
print("Length of input string: ", a)
b = int(a/4)

print("First half of half of the given string of len ", a, "is: ", str1[0:b])
```

```
Length of input string:  64
First half of half of the given string of len  64 is:  The judge senten
```

[13]:
```
# Write a program to get a single string from two given strings separated by a␣
 ↪space and swap the first two characters of each string.

a = input("Enter first string: ")
b = input("Enter second string: ")

c = a + " " + b
print("Combined string separated by space: ", c)

# Since strings are immutable, new string has to be created to make the changes
a_new = b[0:2] + a[2:]
b_new = a[0:2] + b[2:]
d = a_new + " " + b_new
print("Combined string separated by space after swapping the first two␣
 ↪characters of each string: ", d)
```

```
Enter first string: Happy
Enter second string: World
Combined string separated by space:  Happy World
Combined string separated by space after swapping the first two characters of
each string:  Woppy Harld
```

[14]:
```
# Write a program to print floating point numbers with no decimal places.
a = float(input("Enter the floating point number: "))
print("Input given: ", a)

print("Same floating point number with no decimal places: ", int(a))
```

```
Enter the floating point number: 1387.12387
Input given:  1387.12387
Same floating point number with no decimal places:  1387
```

[ ]: