Lagrange's Theorem and Cosets

October 21, 2021

1 Lagrange's Theorem

Binary operation: Let A be a non-empty set. Then a binary operation $*$ on A is a function $* : A \times A \to A$ defined by $*(a, b) = a*b$.

In other words $*$ is binary operation on a set A if $a*b\ A,\ a, b\ A$. In this case we say that A is closed under $*$ or closure property holds in A w.r.t $*$.

Group: A non-empty set G with a binary operation $*$, denoted by $(G, *)$, is said to be a group if the following properties (or axioms) are satisfied. 1. Closure property: $a*b\ G$ for any $a, b\ G$. 2. Associate property: $(a*b)*c = a*(b*c)$ for any $a, b, c\ G$. 3. Existence of identity element: There exists e in G such that $a*e = e*a = a$ for all $a\ G$. 4. Existence of inverse element: For any a in G there exists $a^(-1)$ in G such that $a*a^(-1) = a^(-1)*a = e$.

Further the group $(G, *)$ is called abelian (or commutative) group if along with above properties $a*b = b*a,\ a, b\ G$, also holds.

1.1 Lagrange's theorem statement

If G is any finite group and H is any subgroup of G, then O(H) divides O(G).

Examples to verify Lagrange's theorem

[1]:
```
from numpy import *

G = array([1, -1, 1j, -1j])
H = array([1, -1])

i = 0
F1 = 0          # Assuming that H is not a subgroup

if H[i]*H[i + 1] == H[i] or H[i]*H[i+1] == H[i + 1] and H[i]*H[i] == H[i] or␣
↪H[i]*H[i] == H[i + 1] and H[i + 1]*H[i + 1] == H[i] or H[i + 1]*H[i + 1] ==␣
↪H[i + 1] :
    print("H is closed under multiplication\n")
    F1 = 1
else:
    print("H is not closed and hence H is not a group.\n")
```

```python
if H[i]*H[i] == H[i] and H[i+1]*H[i] == H[i+1]:
    e = H[i]
    print(f'e = {H[i]} is an unique identity element')
    F1 = 1
elif H[i]*H[i+1] == H[i] and H[i+1]*H[i+1] == H[i+1]:
    print(f'e = {H[i+1]} is an unique identity element')
    F1 = 1
else:
    print("No identity element exists.\n H is not a Group.")



if H[i]*H[i] == H[i] and H[i+1]*H[i] == H[i+1]:
    e = H[i]
    print(f'e = {H[i]} is an unique identity element')
    F1 = 1
elif H[i]*H[i+1] == H[i] and H[i+1]*H[i+1] == H[i+1]:
    print(f'e = {H[i+1]} is an unique identity element')
    F1 = 1
else:
    print("No identity element exists.\n H is not a Group.")



if F1 == 1:
    print("H satisfies all the three axioms under multiplication \n");
    print("Hence H is a group \n")
    print("H is subset of group G implies H is subgroup of G \n");
else:
    print("H is not a subgroup of G.")

print("Lagranges theorem: O(H) divides O(G) \n");



n=len(G);
m=len(H);
print(f"order of G = {n}\n");
print(f"order of H = {m} \n");
k = int(n / m);

if mod(n,m) == 0:
    print(f"O(G)/O(H) = {k}")
    print("\nHence Lagranges theorem holds.\n");
else:
    print(f"{k} is not divisor \n");
```

```
    print("Hence Lagranges theorem does not hold.\n");
```

```
H is closed under multiplication

e = 1 is an unique identity element
e = 1 is an unique identity element
H satisfies all the three axioms under multiplication

Hence H is a group

H is subset of group G implies H is subgroup of G

Lagranges theorem: O(H) divides O(G)

order of G = 4

order of H = 2

O(G)/O(H) = 2

Hence Lagranges theorem holds.
```

Examples for finding right and left coset and the index of a group.

[2]:
```
from numpy import *
Z = arange(0, 9);
H = array([0, 3, 6])
m=len(Z);
n=len(H);
I = int(m/n);
print(f"Index of a group is {I} \n");

YL = empty((I, n))
for i in arange(n):
    YL[0,i] = H[i];
for i in arange(1, I):
    for j in arange(0, n):
        YL[i, j] = mod(i + H[j], m)
        if YL[i,j] == YL[0, 0] or YL[i,j] == YL[0, 1] or YL[i,j] == YL[0, n-1]:
            break
print(f'The distinct left cosets are \n {YL}');

YR = empty((I, n))
for i in arange(n):
    YR[0,i] = H[i];
for i in arange(1, I):
    for j in arange(0, n):
```

```python
        YR[i, j] = mod(H[j] + i, m)
        if YR[i,j] == YR[0, 0] or YR[i,j] == YR[0, 1] or YR[i,j] == YR[0, n-1]:
            break
print(f'The distinct right cosets are \n {YR}');
```

```
Index of a group is 3

The distinct left cosets are
 [[0. 3. 6.]
 [1. 4. 7.]
 [2. 5. 8.]]
The distinct right cosets are
 [[0. 3. 6.]
 [1. 4. 7.]
 [2. 5. 8.]]
```