Raw DATA

```
n=input("Enter the number of obs :- ");
list=[];
for i=1:n
   n1=input("Enter the number :- ");
   list=[list,n1];
end
l1=sort(list);
if mod(n,2)==0
   fprintf("N is even !!");
   median1=n/2;
   median2=(n/2)+1;
   m1=list(median1);
   m2=list(median2);
   median=(m1+m2)/2;
   fprintf("Median is :- %d\n",median);
else
   fprintf("N is odd !!");
   median1=(n+1)/2;
   m1=list(median1);
   median=m1;
   fprintf("Median is :- %d\n",median);
end
```

Frequency Distribution

```
n12=input("Enter the number of observations :- ");
data=[];
freq=[];
for i=1:n12
   n1=input("Enter the number :- ");
   data=[data,n1];
end
for i=1:n12
   n2=input("Enter the number :- ");
   freq=[freq,n2];
end
cum_freq = cumsum(freq);
n = sum(freq);
if mod(n, 2) == 0
   n_over_2 = n / 2;
   median_class_idx = find(cum_freq >= n_over_2, 1);
   median_class = data(median_class_idx);
   prev_cum_freq = cum_freq(median_class_idx) - freq(median_class_idx);
```

```
   freq_median_class = freq(median_class_idx);
   median = median_class + ((n_over_2 - prev_cum_freq) / freq_median_class) *
(data(median_class_idx+1) - median_class);
else
   n_plus_1_over_2 = (n + 1) / 2;
   median_class_idx = find(cum_freq >= n_plus_1_over_2, 1);
   median_class = data(median_class_idx);
   prev_cum_freq = cum_freq(median_class_idx) - freq(median_class_idx);
   median = median_class + ((n_plus_1_over_2 - prev_cum_freq) /
freq(median_class_idx)) * (data(median_class_idx+1) - median_class);
end
disp(['The median of the grouped data is: ', num2str(median)]);
```