Problem Definition: Assume Suitable data set of 100 numeric values. The Hash Function is h(x)=x mod 128. The Bucket size is 6 locations .The Gain Factor Varies between 2-5 . Write Code & generate collision table form different gain factors between 2-5. If new component of size 20 is added in data set illustrate the storage snapshot.

------> Here we have to perform Extendible hashing where g varied from 2-5 and size of bucket is 6.Extendible hashing is an approach which is dynamic i.e. it allows insertions and deletions to occur without resulting in poor performance after many occurrences of these operations. In my solution I had maded a class bucket which can contain 6 keys and a hash class which has an array list with the directory names and each of these directories have reference to the object of bucket.

Technology Stack : Problem Solved In Net-Beans IDE V8.2,Using Java JDK 8.

Classes And Functions:

Classes-

1. extendiable
2. Hashing
3. Bucket

Functions In each Class-

1. **Extendiable- public static void main :**This Is main Class and contains static main and works as driver for the whole program.
2. **Hashing -**
    a. **Insertion:** It Inserts the 20 integers at last .It helps in converting the last 20 numbers in binary and then calling remaining functions to add these numbers in hash table.
    b. **calculateHash_binary:**It calculates x mod 128 and then converts them to binary and adds them to array listy which will be used to add data in buckets.
    c. **masterdirectoryinitial:** It initializes the initial directory list.
    d. **masterdirdoubling:**It doubles the directory when there is a need for doubling the directory i.e. number of buckets in case of collisions.
    e. **addingdata:**It actually adds the data and calls the additem method of bucket class. It also calls the realltoment and realltment2 whichever is required.
    f. **reallotment:**it is called when it is needed to double the directory.
    g. **reallotment2:** it is called when directory doubling is not required and only existing buckets need to be filled.
    h. **display:**called to display the snapshot of the hash tables.
3. **Bucket-**
    a. make-bucket:called to initialize the bucket with suitable flags.
    b. displaybucket:displays each data of bucket.
    c. additem:this method actually adds the data in bucket.

Key Functionality: This project aims on working of extendiable hashing and how large data can be handled easily with this type of hashing.