

Q1)

Pseudo Code

- 1) Initialize T for Test cases,
- 2) Starting a loop for ever single T ,
- 3) Initialize flag , string ,
- 4) Check the value of string length , if length in less than 3 than print DYNAMIC,
- 5) Initialize a map which take the values of string.
- 6) Initialize a vector ,
- 7) Perform a loop on map and put map->second value in vector,
- 8) Then applying sorting on vector
- 9) Now perform a check on if $\text{vector}[3] \neq \text{vector}[2] + \text{vector}[1]$, then swap the value of $\text{vector}[0]$, $\text{vector}[1]$,
- 10) Perform a loop which iterate on vector and if $\text{vector}[i] \neq \text{vector}[i-1] + \text{vector}[i-2]$ then increase the value of flag by 1 and break.
- 11) Now check the value of flag if flag = 0 then print DYNAMIC else Not .

Implementation

```
#include<iostream>
#include <bits/stdc++.h>
using namespace std;
#define ll long long

int main() {
    ll T;
    cin>>T;
    while(T--)
    {
        ll flag=0;
        string S;
        cin>>S;
        if(S.length()<3)
        {
            cout<<"Dynamic"<<endl;
            continue;
        }
        map<char, ll> mVar;
        vector<ll> array1;
        ll tmp[100001]={0};
        for(ll i=0;i<S.length();i++)
        {
            mVar[S[i]]++;
        }
    }
}
```

```

    }
    for(auto var:mVar)
    {
        ll ins=var.second;
        array1.push_back(ins);
    }
    sort(array1.begin(),array1.end());
    if(array1[3]!=array1[2]+array1[1])
    {
        swap(array1[0],array1[1]);
    }
    for(ll i=2;i<array1.size();i++)
    {
        if(array1[i]!=array1[i-1]+array1[i-2])
        {
            flag++;
            break;
        }
    }
}

```

```

if(flag==0)
{
    cout<<"Dynamic"<<endl;
}
else
{
    cout<<"Not"<<endl;
}
}
return 0;
}

```