Attempt 1 : partially correct

    std::map<char,int> counter;

    for (int i=0; i < s.length(); i++){

        counter[s[i]]++;

    }

    int max = counter.begin()->second;

    int min = counter.begin()->second;

    for (auto pair : counter) {

      if (max<pair.second){

        max=pair.second;

      }

      if (min>pair.second){

        min=pair.second;

      }

    }

    cout<< max<<' ' <<min;

    if (max-min>1){

        return "NO";

    }

    else{

        return "YES";

    }

Attempt 2

    std::map<char,int> counter;

    for (int i=0; i < s.length(); i++){

        counter[s[i]]++;

    }

    //int max = counter.begin()->second;

    std::map<int,int> res;

    for (auto pair : counter) {

     res[pair.second]++;

    }

    for (auto pair : res) {

     cout<<pair.first<<':'<<pair.second<<'\n';

    }

    return s;

attempt 3 correct

const char \* s\_chars = s.c\_str();

vector<int> occur(26);

for(int i = 0; i < s.length(); i++ ){

occur[s\_chars[i] - 'a']++;

}

int max\_occur =-1;

bool removed\_char = false;

for(int i =0; i < 26; i++){

if(occur[i] == 0){

continue;

}else if(max\_occur == -1){

max\_occur = occur[i];

continue;

}else if(occur[i] == max\_occur){

continue;

}else if(!removed\_char && (occur[i] == max\_occur + 1 || occur[i] == 1)){

removed\_char = !removed\_char;

continue;

}else{

return "NO";

}

}

return "YES";