Plus one

由题意，给你一个数组，其中每一位对应一个非负数的每一位，现在需要对其进行加1运算，并也用数组的形式返回结果值：

第一次做的代码如下：

class Solution {

public:

vector<int> plusOne(vector<int>& digits) {

vector<int> temp;

int leg;

leg=digits.size();

if(leg==0)

{

digits.push\_back(1);

return digits;

}

int flag=1;

int end=-1;

for(int i=leg-1;i>=0;i--)

{

if(digits[i]==9&&flag==1)

{

temp.push\_back(0);

flag=1;

}

else

{

end=i;

temp.push\_back(digits[i]+1);

break;

}

}

if(end==-1)

{

digits.clear();

digits.push\_back(1);

for(int i=1;i<leg+1;i++)

digits.push\_back(0);

return digits;

}

else

{

for(int i=end-1;i>=0;i--)

temp.push\_back(digits[i]);

digits.clear();

for(int j=temp.size()-1;j>=0;j--)

digits.push\_back(temp[j]);

return digits;

}

}

};

第二次学会rbegin()用法后，代码如下：

class Solution {

public:

vector<int> plusOne(vector<int>& digits) {

vector<int> result;

int l;

l=digits.size();

if(l==0){

result.push\_back(1);

return result;

}

int flag=0;

int count=0;

if(digits[l-1]==9){

flag=1;

for(int i=l-1;i>=0;i--){

if(digits[i]==9&&flag==1){

result.push\_back(0);

count++;

}

else if(flag==1){

result.push\_back(digits[i]+1);

flag=0;

}

else{

result.push\_back(digits[i]);

}

}

if(count==l){

result.push\_back(1);

return vector<int>(result.rbegin(),result.rend());

}

return vector<int>(result.rbegin(),result.rend());

}

else{

for(int j=0;j<l-1;j++){

result.push\_back(digits[j]);

}

result.push\_back(digits[l-1]+1);

return result;

}

}

};