class Solution {

public:

vector<string> letterCombinations(string digits) {

vector<string>res;

if(digits.empty()) return res;

vector<string>letter({"", "", "abc", "def", "ghi", "jkl", "mno", "pqrs", "tuv", "wxyz"});

string path = "";

DFS(digits, 0, path, res, letter);

return res;

}

void DFS(string digits, int pos, string& path, vector<string>& res, vector<string>& letter){

if(pos == digits.size()){

res.push\_back(path);

return;

}

for(auto c: letter[digits[pos] - '0']){

path.push\_back(c);

DFS(digits, pos + 1, path, res, letter);

path.pop\_back();

}

}

};