

二十四点游戏的深度搜索算法demo

```
def solve24(nums):
    def dfs(nums, path):
        if len(nums) == 1:
            if abs(nums[0] - 24) < 1e-6:
                result.append(path)
                return True
            return False

        for i in range(len(nums)):
            for j in range(len(nums)):
                if i != j:
                    next_nums = [nums[k] for k in range(len(nums)) if k != i and k != j]
                    if dfs(next_nums + [nums[i] + nums[j]], path + f"{nums[i]} + {nums[j]} = {nums[i] + nums[j]}"):
                        return True
                    if dfs(next_nums + [nums[i] - nums[j]], path + f"{nums[i]} - {nums[j]} = {nums[i] - nums[j]}"):
                        return True
                    if dfs(next_nums + [nums[i] * nums[j]], path + f"{nums[i]} * {nums[j]} = {nums[i] * nums[j]}"):
                        return True
                    if nums[j] != 0 and dfs(next_nums + [nums[i] / nums[j]], path + f"{nums[i]} / {nums[j]} = {nums[i] / nums[j]}"):
                        return True
        return False

    result = []
    if dfs(nums, ""):
        return result[0]
    else:
        return "无法得到24!"

# 输入数字
nums = [20, 20, 2, 15]

# 解决24点游戏并输出完整计算过程
solution = solve24(nums)
print(solution)
```