NTUEE algorithm PA2 report

B09901081 施伯儒

data structures: Array , vector(c++ stl)

findings and some problems:

1. **Bottom-up method will lead to TLE,since we don't need to calculate every case.**
2. **You need to use Top-down method with memorization to boost your speed.**
3. **Array is faster than vector when it comes to inserting many elements.**
4. **You should carefully use the 2D array otherwise it might be over the limited stack size**
5. **Sometimes when your program got killed on the server doesn't mean yours is wrong. Too many users on the server can lead to this result.**
6. **Using ulimit to set the stack size can only be set once.If you want to set it again,you can only set it lower.**

solutions to problems:

1. **I use recursive call but carefully check whether this (i,j) is calculated before, which can boost my speed.**
2. **I try to use Hash Table to minimize the memory, but I need to redefined Hash function in unordered_map(c++ stl), otherwise it cannot support multiple keys. Perhaps my hash function design is too bad, it turned out to be a disaster,which slows down my program and lead to many unpredictable problems(ex:collision).**
3. **To minimize memory, I only use a 2D array ,but in a clever way, I can remember three things in a single Integer value.Therefore I only use one 2D array,which is a comparably acceptable result.**

⇒sol:

```
int **answer;
    // answer[i][j]%10 = 0 means unvisited
    // answer[i][j]%10 = 1 means visited
    // answer[i][j]%100 means which case is (i,j)
    // answer[i][j]/100 means optimal number of chords
```