2017.8.1拼多多在线笔试题

最近在狼厂实习中，很久没做题了。秋招第一发， 拼多多。。。

四个简单题，看到有些人竟然觉得难？ 我来降一发自己的RP，这题目觉得难的，如果你拿到比我好的Offer，我是不服气的。。

四个题。。。其实我也就写了40分钟吧。。不过最后也没有满分， 390/400， 第三题不知道为嘛一直有10分过不了。。

更一下, 刚刚好像发现第三题。。。这个>号， 我写的是>= ....? 可是我看题目好像是 >= 呀。。。

第一题:

要求时间复杂度O(n)， 空间复杂度O(1)。

那么其实答案有两种情况，最大的三个数相乘 || 最小的两个数 \* 最大的数。

时间复杂度O(n)，瞬间想到时间复杂度O(n)求k大的经典算法，分治法！

#include <cstdio>

#include <iostream>

#include <string>

#include <algorithm>

using namespace std;

const int N = 1e6 + 10;

long long a[N];

int k;

int partition(int l,int r) {

while(l != r)

{

while(a[r] >= a[l] && r > l)

r--;

if(l == r)

break;

swap(a[r],a[l]);

while(a[l] < a[r] && r > l)

l++;

if(l < r)

swap(a[r],a[l]);

}

return l;

}

long long solve(int l,int r) {

int now = partition(l,r);

if(k < now)

return solve(l,now-1);

else if(k > now)

return solve(now+1,r);

else

return a[now];

}

int main() {

int n;

while(~scanf("%d", &n)) {

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

scanf("%lld", &a[i]);

}

k = n - 1;

long long x1 = solve(0, n-1);

k = n - 2;

long long x2 = solve(0, n-2);

k = n - 3;

long long x3 = solve(0, n-3);

long long Ans = x1 \* x2 \* x3;

if(n > 3) {

k = 0;

long long y1 = solve(0, n-1);

k = 1;

long long y2 = solve(0, n-2);

Ans = max(Ans, y1\*y2\*x1);

}

printf("%lld\n", Ans);

}

return 0;

}

第二题:

大数相乘，模板题， 找了个模板。。。

#include <iostream>

#include <cstring>

#include <string>

#include <algorithm>

using namespace std;

const int N = 1e5 + 10;

string c1, c2;

int a[N], b[N], r[N];

void solve(int a[], int b[], int la, int lb) {

int i, j;

for(i = 0; i != N; i++) r[i] = 0;

for(i = 0; i != la; i++)

{

for(j = 0; j != lb; j++)

{

int k = i + j;

r[k] += a[i] \* b[j];

while(r[k] > 9)

{

r[k + 1] += r[k] / 10;

r[k] %= 10;

k++;

}

}

}

int l = la + lb - 1;

while(r[l] == 0 && l > 0) l--;

for(int i = l; i >= 0; i--) cout << r[i];

cout << endl;

}

int main() {

while(cin >> c1 >> c2)

{

int la = c1.size(), lb = c2.size();

for(int i = 0; i != la; i++)

a[i] = (int)(c1[la - i - 1] - '0');

for(int i = 0; i != lb; i++)

b[i] = (int)(c2[lb - i - 1] - '0');

solve(a, b, la, lb);

}

return 0;

}

第三题：

贪心啊， 我是按照 尽量满足最小人的需求来贪心的。。。一直90%?

有人是用尽量使用掉最大的巧克力来贪的，100%， 来个反例好不好？

#include <cstdio>

#include <iostream>

#include <string>

#include <algorithm>

using namespace std;

const int N = 3e6 + 10;

long long w[N], h[N];

int main() {

int n, m;

while(~scanf("%d", &n)) {

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

scanf("%lld", &h[i]);

}

scanf("%d", &m);

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

scanf("%lld", &w[i]);

}

sort(h, h + n);

sort(w, w + m);

int Ans = 0;

for(int i = 0, j = 0; i < n && j < m; ) {

if(w[j] >= h[i]) {

++Ans;

++i, ++j;

}

else {

++j;

}

}

printf("%d\n", Ans);

}

return 0;

}

第四题：

迷宫问题， 有趣的是多了一把钥匙。。。

一看门不超过10个。。。M, N <=100...想了想状态数。。。直接状态压缩吧。。

之后就是一个非常暴力可耻的状态压缩bfs。。。然后就一发AC了。。

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89 | #include <cstdio>  #include <iostream>  #include <string>  #include <queue>  #include <map>  #include <algorithm>  using namespace std;  const int N = 110;  char mz[N][N];  bool vis[N][N][N\*10];  int fx[4] = {0, 0, 1, -1};  int fy[4] = {1, -1, 0, 0};  int m, n;  map<char, int> key;    struct node {      int x, y, cnt, sta;      node():cnt(0), sta(0) {}  };  queue<node> que;    int bfs(int sx, int sy, int ex, int ey) {      while(!que.empty()) que.pop();      node tmp;      tmp.x = sx, tmp.y = sy;      que.push(tmp);        while(!que.empty()) {          node p = que.front();          if(p.x == ex && p.y == ey) {              return p.cnt;          }          que.pop();            for(int i = 0; i < 4; ++i) {              int newx = p.x + fx[i];              int newy = p.y + fy[i];              if(newx < 0 || newx >= m || newy < 0 || newy >= n) continue;              if(mz[newx][newy] == '0') continue;              int sta = p.sta;              if(mz[p.x][p.y] >= 'a' && mz[p.x][p.y] <= 'z') {                  sta |= (1<<key[mz[p.x][p.y]]);              }              if(vis[newx][newy][sta]) continue;              if(mz[newx][newy] >= 'A' && mz[newx][newy] <= 'Z') {                  if((sta & (1<<(key[mz[newx][newy] - 'A' + 'a'])))== 0) {                      continue;                  }              }              vis[newx][newy][sta] = true;              tmp.x = newx, tmp.y = newy, tmp.cnt = p.cnt + 1, tmp.sta = sta;              que.push(tmp);          }      }      return -1;  }    int main() {      while(~scanf("%d %d", &m, &n)) {          int sx, sy, ex, ey;          int cnt = 0;          for(int i = 0; i < m; ++i) {              scanf("%s", mz[i]);              for(int j = 0; j < n; ++j) {                  if(mz[i][j] == '2') {                      sx = i, sy = j;                  }                  if(mz[i][j] == '3') {                      ex = i, ey = j;                  }                  if(mz[i][j] >= 'a' && mz[i][j] <= 'z') {                      key[mz[i][j]] = cnt++;                  }              }          }            for(int i = 0; i < m; ++i) {              for(int j = 0; j < n; ++j) {                  for(int s = 0; s < (1<<cnt); ++s) {                      vis[i][j][s] = false;                  }              }          }            int Ans = bfs(sx, sy, ex, ey);          printf("%d\n", Ans);      }      return 0;  } |