今晚拼多多编程题代码

第一题,先找出最大的三个数,记这个为ans1,再找出最小的两个数,将他们和上一步求出的最大的数相乘,记为ans2,然后答案是max(ans1,ans3),注意一下结果会爆int

|  |  |
| --- | --- |
| 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  90 | #include<stdio.h>  #include<string.h>  #include<algorithm>  #include<queue>  using namespace std;    long long n,a[10000000];    int main()  {      scanf("%lld",&n);      for(int i=0;i<n;i++)      {          scanf("%lld",a+i);      }      long long  ans = a[0]\*a[1]\*a[2];      long long  a1,a2,a3;      a1=a[0];      a2=a[1];      a3=a[2];      if(a2<a3)      {          swap(a2,a3);      }      if(a1<a3)      {          swap(a1,a3);      }      if(a1<a2)      {          swap(a1,a2);      }      for(int i=3;i<n;i++)      {          if(a[i]>a3)          {              if(a[i]>a2)              {                  a3=a2;                  if(a[i]>a1)                  {                      a2=a1;                      a1=a[i];                  }                  else                  {                      a2=a[i];                  }              }              else              {                  a3=a[i];              }          }      }      ans = max(ans,a1\*a2\*a3);      if(a[0]==a1) {          a2 = a[1];          a3 = a[2];      }      else if(a[1]==a1)      {          a2=a[0];          a3=a[2];      }      else      {          a2=a[0];          a3=a[1];      }      if(a2>a3) swap(a2,a3);      for(int i=2;i<n;i++)      {          if(a[i]<a3)          {              if(a[i]<a2)              {                  a3=a2;                  a2=a[i];              }              else              {                  a3=a[i];              }          }      }      ans = max(ans,a1\*a2\*a3);      printf("%lld\n",ans);      return 0;  } |

第二题,模拟一下乘法操作就可以了

|  |  |
| --- | --- |
| 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 | #include<stdio.h>  #include<string.h>  #include<algorithm>  #include<queue>  using namespace std;    int cnt=0;  char t[1000000],a[1000000],b[1000000],ans[10000000];      int main()  {      memset(a,0,sizeof(a));      memset(b,0,sizeof(b));      memset(ans,0,sizeof(ans));      scanf("%s",t);      int lena=strlen(t);      for(int i=lena-1,j=0;i>=0;i--,j++)      {          a[j]=t[i]-'0';      }      scanf("%s",t);      int lenb=strlen(t);      for(int i=lenb-1,j=0;i>=0;i--,j++)      {          b[j]=t[i]-'0';      }      //printf("%s %s\n",a,b);      int now;      int cnt=-1;      for(int i=0;i<lena;i++)      {          now=i;          int jin = 0;          for(int j=0;j<lenb;j++)          {              int tmp=a[i]\*b[j]+jin+ans[now];              //printf("%d %d %d\n",tmp,tmp%10,tmp/10);              ans[now] = tmp%10;              jin = tmp / 10;              now++;              cnt = max(now,cnt);          }          while(jin!=0)          {              int tmp=jin+ans[now];              ans[now] = tmp%10;              jin = tmp/10;              cnt = max(now,cnt);          }      }      while(ans[cnt]==0&&cnt!=0) cnt--;      for(int i=cnt;i>=0;i--)          printf("%c",ans[i]+'0');      puts("");      return 0;  } |

第三题,贪心,对于每一个h,找一个最合适的w,将输入的两个数组排序,然后用两个指针,移动一下就好了

|  |  |
| --- | --- |
| 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 | #include<stdio.h>  #include<string.h>  #include<algorithm>  #include<queue>  using namespace std;  int n,m,w[10000],h[10000];    int main()  {      scanf("%d",&n);      for(int i=0;i<n;i++) scanf("%d",h+i);      scanf("%d",&m);      for(int i=0;i<m;i++) scanf("%d",w+i);      sort(h,h+n);      sort(w,w+m);      int cnt=0;      int j=0;      for(int i=0;i<n;i++)      {          while(j<m&&w[j]<h[i])          {              j++;          }          if(j==m) break;          j++;          cnt++;      }      printf("%d\n",cnt);      return 0;  } |

第四题,bfs+状态压缩,比普通的dfs就是多几把钥匙,问题中最多为10把,然后可以用一个数来代表有没有某一把钥匙,即这个数的二进制表示时第i位为1代表有第a+i把钥匙,否则没有,相当于迷宫每一个点有1024个状态

|  |  |
| --- | --- |
| 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 | #include<stdio.h>  #include<string.h>  #include<algorithm>  #include<queue>  using namespace std;  int n,m,sx,sy,ex,ey;  char g[1024][1024];  int use[120][120][1400];  int dx[]={0,0,1,-1};  int dy[]={1,-1,0,0};  struct node  {      int x,y,k;  };    int bfs()  {      memset(use,0xff,sizeof(use));      queue<node>q;      node t;      t.x=sx;      t.y=sy;      t.k=0;      use[t.x][t.y][t.k]=0;      q.push(t);      while(!q.empty())      {          t = q.front();          q.pop();          //printf("%d %d %d %d\n",t.x,t.y,t.k,use[t.x][t.y][t.k]);          if(t.x==ex&&t.y==ey) return use[t.x][t.y][t.k];          for(int i=0;i<4;i++)          {              node k;              k.x = t.x + dx[i];              k.y = t.y + dy[i];              k.k = t.k;              if(k.x<0||k.x>=n||k.y<0||k.y>=m||g[k.x][k.y]=='0') continue;              if(g[k.x][k.y]>='a'&&g[k.x][k.y]<='z')              {                  k.k = k.k|(1<<(g[k.x][k.y]-'a'));              }              if(g[k.x][k.y]>='A'&&g[k.x][k.y]<='Z')              {                  int p = k.k&(1<<(g[k.x][k.y]-'A'));                  //printf("%d %d %d\n",k.k,1<<(g[k.x][k.y]-'A'),p);                  if(p==0) continue;              }              if(use[k.x][k.y][k.k]==-1||use[k.x][k.y][k.k]>use[t.x][t.y][t.k]+1)              {                  use[k.x][k.y][k.k]=use[t.x][t.y][t.k]+1;                  q.push(k);              }          }      }      return -1;  }  int main()  {      scanf("%d%d",&n,&m);      for(int i=0;i<n;i++)      {          scanf("%s",g[i]);          for(int j=0;j<m;j++)          {              if(g[i][j]=='2') {sx=i;sy=j;}              if(g[i][j]=='3') {ex=i;ey=j;}          }      }      printf("%d\n",bfs());      return 0;  } |