# Yuesefu

2886

#include <iostream>

#include <cstdio>

using namespace std;

#define maxn 500010

#define mid ((l+r)>>1)

#define tmp (st<<1)

#define lson l,mid,tmp

#define rson mid+1,r,tmp|1

int sum[maxn<<2];

#define mod sum[1]

struct N{

char s[100];

int num;

void in(){

scanf("%s%d",s,&num);

}

}my[maxn];

void build(int l,int r,int st){

sum[st]=r-l+1;

if(l==r)return ;

build(lson);

build(rson);

}

int update(int c,int l,int r,int st){

sum[st]--;

if(l==r)return r;

if(c<=sum[tmp])return update(c,lson);

return update(c-sum[tmp],rson);

}

int antiprime[]={

1, 2, 4, 6, 12,

24, 36, 48, 60, 120,

180, 240, 360, 720, 840,

1260, 1680, 2520, 5040, 7560,

10080, 15120, 20160, 25200, 27720,

45360, 50400, 55440, 83160, 110880,

166320, 221760, 277200, 332640, 498960};

int factorNum[]={

1, 2, 3, 4, 6,

8, 9, 10, 12, 16,

18, 20, 24, 30, 32,

36, 40, 48, 60, 64,

72, 80, 84, 90, 96,

100,108,120,128,144,

160,168,180,192,200};

int main(){

int n,k;

while(scanf("%d%d",&n,&k)!=EOF){

build(1,n,1);

for(int i=1;i<=n;++i)my[i].in();

int ans=0;

while(ans<35&&antiprime[ans]<=n)ans++;

ans--;

int pos=0;

my[pos].num=0;

for(int i=0;i<antiprime[ans];i++){

if(my[pos].num>0)

k=((k-1+my[pos].num-1)%mod+mod)%mod+1;

else

k=((k-1+my[pos].num)%mod+mod)%mod+1;

pos=update(k,1,n,1);

}

printf("%s %d\n",my[pos].s,factorNum[ans]);

}

return 0;

}