

```
#include<bits/stdc++.h>
```

```
using namespace std;
```

```
struct matrix
```

```
{
```

```
    long long v[4][4];
```

```
    long long row,col;
```

```
};
```

```
long long mod;
```

```
matrix multi(matrix a, matrix b)
```

```
{
```

```
    assert(a.col == b.row);
```

```
    matrix r;
```

```
    r.row = a.row;
```

```
    r.col = b.col;
```

```
    for(long long i=0; i<r.row; i++)
```

```
        for(long long j=0; j<r.col; j++)
```

```
        {
```

```
            long long sum = 0;
```

```
            for(long long k=0; k<a.col; k++)
```

```
            {
```

```
                sum+=(a.v[i][k] * b.v[k][j]);
```

```
                sum%=mod;
```

```
            }
```

```
            r.v[i][j] = sum;
```

```
        }
```

```
    return r;
```

```
}
```

```
matrix power(matrix mat,long long p)
```

```
{
```

```
    assert(p >=1 );
```

```
    if(p==1) return mat;
```

```

    if(p%2 == 1)
        return multi(mat,power(mat,p-1));

    matrix ret = power(mat,p/2);
    ret = multi(ret,ret);
    return ret;
}

int main()
{
    long long n,m,t,w =0;

    cin>>t;
    while(t--)
    {
        long long a,b;
        cin>>a>>b>>n>>m;

        mod = 1;
        while(m--) mod*=10;

        matrix mt;

        mt.row = mt.col = 2;

        mt.v[0][0] =1, mt.v[0][1] = 1;
        mt.v[1][0] = 1,mt.v[1][1] = 0;

        printf("Case %lld: ",++w);

        if(n<3){
            if(n==0) cout<< a%mod<<endl;
            if(n==1) cout<< b%mod<<endl;
            if(n==2) cout<< (a + b) % mod<<endl;
        }
        else
        {
            mt = power(mt,n-2);

            long long ans = (a + b) * mt.v[0][0] + b * mt.v[0][1];

            ans%=mod;

            cout<<ans<<endl;
        }
    }
}

```

```
}
```

```
}
```

```
return 0;
```

```
}
```