/\*

PROG: damage

LANG: C++

ID: hayk.sa1

\*/

#include <stdio.h>

int e[100010][2];

int \*g[30010], deg[30010];

int s[30010];

int t[30010];

int st[30010];

void dfs()

{

int p, l, i;

l = 1;

t[0] = 1;

st[0] = 0;

while (l)

{

p = st[--l];

for (i = 0; i < deg[p]; i++)

if (t[g[p][i]] == 0)

{

t[g[p][i]] = 1;

st[l++] = g[p][i];

}

}

}

int main()

{

freopen("damage.in", "r", stdin);

freopen("damage.out", "w", stdout);

int n, m, k, s, i, j;

scanf("%d%d%d", &n, &m, &k);

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

{

scanf("%d%d", e[i], e[i]+1);

if (e[i][0] != e[i][1])

{

deg[--e[i][0]]++;

deg[--e[i][1]]++;

}

}

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

{

g[i] = new int[deg[i]+1];

deg[i] = 0;

}

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

if (e[i][0] != e[i][1])

{

g[e[i][0]][deg[e[i][0]]++] = e[i][1];

g[e[i][1]][deg[e[i][1]]++] = e[i][0];

}

for (i = 0; i < k; i++)

{

scanf("%d", &s);

for (j = 0; j < deg[s-1]; j++)

t[g[s-1][j]] = 2;

}

if (t[0] == 2)

{

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

return 0;

}

dfs();

m = 0;

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

m += (t[i] != 1);

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

return 0;

}