/\*

PROG: ski

LANG: C++

ID: hayk.sa1

\*/

#include

#include

using namespace std;

#define MAXA 100

#define MAXT 10000

struct lesson

{

int m, l, a;

} l[110];

bool operator < (lesson a, lesson b)

{

return a.m < b.m;

}

int a[110];

int d[110][110];

int main()

{

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

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

int ans, t, n, m, p, q, i, j, k;

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

l[0].m = 0;

l[0].l = 0;

l[0].a = 1;

for (i = 1; i <= n; i++)

{

scanf("%d%d%d", &l[i].m, &l[i].l, &l[i].a);

if (l[i].m+l[i].l > t)

{

i--;

n--;

}

}

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

a[i] = MAXT+1;

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

{

scanf("%d%d", &p, &q);

for (j = p; j <= MAXA; j++)

if (a[j] > q)

a[j] = q;

}

m = t/a[1];

sort(l, l+n+1);

memset(d, -1, sizeof(d));

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

d[i][1] = 0;

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

for (j = 1; j <= MAXA; j++)

if (d[i][j] != -1)

{

p = a[j];

for (k = i+1; k <= n; k++)

if (d[k][j] < d[i][j]+(l[k].m-l[i].m)/p)

d[k][j] = d[i][j]+(l[k].m-l[i].m)/p;

if (l[i].a > j)

{

p = a[l[i].a];

for (k = i+1; k <= n; k++)

if (l[k].m > l[i].m+l[i].l &&

d[k][l[i].a] < d[i][j]+(l[k].m-l[i].m-l[i].l)/p)

d[k][l[i].a] = d[i][j]+(l[k].m-l[i].m-l[i].l)/p;

}

if (m < d[i][j]+(t-l[i].m)/a[j])

m = d[i][j]+(t-l[i].m)/a[j];

if (l[i].a > j && m < d[i][j]+(t-l[i].m-l[i].l)/a[l[i].a])

m = d[i][j]+(t-l[i].m-l[i].l)/a[l[i].a];

}

for (i = 1; i <= MAXA; i++)

if (d[n][i] != -1)

{

p = a[i];

if (m < d[n][i]+(t-l[n].m)/p)

m = d[n][i]+(t-l[n].m)/p;

if (l[n].a > i)

{

p = a[l[n].a];

if (m < d[n][i]+(t-l[n].m-l[n].l)/p)

m = d[n][i]+(t-l[n].m-l[n].l)/p;

}

}

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

return 0;

}