#include <stdio.h>

#include <string.h>

#define inf 1000000000

#define swap(a, b) {a^=b;b^=a;a^=b;}

int n;

int e[10010][3];

int dp[1010][1010];

int \*g[1010], \*w[1010], deg[1010];

int m;

int d[1010];

int h[1010], pl[1010];

void up(int p)

{

while (p > 1 && d[h[p]] < d[h[p>>1]])

{

swap(h[p], h[p>>1]);

swap(pl[h[p]], pl[h[p>>1]]);

p >>= 1;

}

}

void down(int p)

{

int l, r, q;

while (p <= m)

{

q = p;

l = p<<1;

r = l|1;

if (l <= m && d[h[l]] < d[h[q]])

q = l;

if (r <= m && d[h[r]] < d[h[q]])

q = r;

if (q == p)

return;

swap(h[p], h[q]);

swap(pl[h[p]], pl[h[q]]);

p = q;

}

}

void dijkstra()

{

int p, i;

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

{

h[++m] = i;

pl[i] = m;

up(m);

}

while (m)

{

p = h[1];

swap(h[1], h[m]);

swap(pl[h[1]], pl[h[m]]);

m--;

down(1);

for (i = deg[p]-1; i >= 0; i--)

if (d[g[p][i]] > d[p] && d[g[p][i]] > w[p][i])

{

if (d[p] > w[p][i])

d[g[p][i]] = d[p];

else

d[g[p][i]] = w[p][i];

up(pl[g[p][i]]);

}

}

}

int main()

{

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

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

int k, i, j, l;

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

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

{

scanf("%d%d%d", e[i], e[i]+1, e[i]+2);

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

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

}

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

{

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

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

deg[i] = 0;

}

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

{

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

w[e[i][0]][deg[e[i][0]]++] = e[i][2];

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

w[e[i][1]][deg[e[i][1]]++] = e[i][2];

}

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

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

dp[i][j] = inf;

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

{

l = 0;

memcpy(d, dp[i], sizeof(dp[i]));

dijkstra();

memcpy(dp[i], d, sizeof(d));

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

for (l = deg[j]-1; l >= 0; l--)

if (dp[i+1][g[j][l]] > dp[i][j])

dp[i+1][g[j][l]] = dp[i][j];

}

memcpy(d, dp[k], sizeof(dp[k]));

dijkstra();

memcpy(dp[k], d, sizeof(d));

l = inf;

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

if (l > dp[i][n-1])

l = dp[i][n-1];

if (l < inf)

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

else

printf("-1\n");

return 0;

}