```python
from pylab import *
import sys, shutil, os
from mpl_toolkits.mplot3d import Axes3D

def read_2D(title,plottitle):
    infile = open(title)
    a = infile.readline().split()
    for i in range(len(a)):
        a[i] = float(a[i])

    u = zeros([len(a),len(a)])
    u[:,0] = a
    j = 1
    for line in infile:
        sline = line.split()
        for i in range(len(sline)):
            a[i] = float(sline[i])
        u[:,j] = a
        j += 1

    fig = figure()
    ax = Axes3D(fig)
    X = linspace(0,1,len(a))
    Y = linspace(0,1,len(a))
    X,Y = meshgrid(X,Y)
    ax.plot_surface(X,Y,u, rstride=1, cstride=1,cmap='cool')

    #ax.contourf(X, Y, u, zdir='z', offset=-10000, cmap=cm.hot)

    xlabel("x"); ylabel("y"); ax.set_zlabel("u(x,y)")
    ax.set_title(plottitle)
    show()

def read_2D_movie(Title,plottitle,subdir,anititle):
    os.chdir("build-main-Desktop-Debug")
    if os.path.isdir(subdir):
        shutil.rmtree(subdir)
    os.mkdir(subdir)
    infile = open(Title)
    os.chdir(subdir)

    u = zeros([10,10])
    counter = 0;
    for line in infile:
        splt = line.split()
        for i in range(10):
            for j in range(10):
                u[j,i] = float(splt[i*10 + j])

        fig = figure()
        ax = Axes3D(fig)
        X = linspace(0,1,10)
        Y = linspace(0,1,10)
        X,Y = meshgrid(X,Y)
        ax.plot_surface(X,Y,u, rstride=1, cstride=1,cmap='hot')
        xlabel("x"); ylabel("y"); ax.set_zlabel("u(x,y)")
        ax.set_title(plottitle)
        savefig("tmp%04d.png"%counter)
        counter += 1
    os.system("mencoder 'mf://tmp*.png' -mf type=png:fps=10 -ovc lavc -lavcopts vcodec=wmv2 -oac copy -o %s"%anititle)
    infile.close()
    os.chdir(os.pardir)
    os.chdir(os.pardir)


def read_1D_MC_bin(Title,Plottitle):
    infile = open(Title)
    X = []
    for line in infile:
        X.append(float(line))
    infile.close()

    #Dividing into Nx different spacial points
    Nx = 20
    dx = 1./Nx; U = zeros(Nx)
    for i in range(len(X)):
        for n in range(Nx-1):
            if X[i] >= n*dx and X[i] < (n+1)*dx:
                U[n] += 1
    print U[0]
    U = U/U[0]
    x = linspace(0,1,len(U))
    plot(x,U,label="Random Walk")
    title(Plottitle)
    xlim(0,1); ylim(0,1)
    xlabel("x")
    ylabel("u")
    legend(loc="upper right")
    show()

def read_1D_MC_movie(Title,Plottitle,subdir,anititle):
    os.chdir("build-main-Desktop-Debug")
    if os.path.isdir(subdir):
        shutil.rmtree(subdir)
    os.mkdir(subdir)
    infile = open(Title)
    os.chdir(subdir)

    Nx = 23; dx = 1./Nx;
    counter = 0
    for line in infile:
        splt = line.split()
        u = zeros(Nx)
        for i in range(len(splt)):
            for n in range(Nx-1):
                if float(splt[i]) >= n*dx and float(splt[i]) < (n+1)*dx:
                    u[n] += 1

        u = u/u[0]
        x = linspace(0,1,len(u))
        plot(x,u,hold=False)
        title(Plottitle)
        xlim(0,1); ylim(0,1)
        xlabel("x")
        ylabel("u")
        savefig("tmp%04d.png"%counter)
        counter += 1
    os.system("mencoder 'mf://tmp*.png' -mf type=png:fps=20 -ovc lavc -lavcopts vcodec=wmv2 -oac copy -o %
s"%anititle)
    infile.close()
    os.chdir(os.pardir)
    os.chdir(os.pardir)

def read_1D(Title,plottitle,Label):
    infile = open(Title)
    u = []
    for line in infile:
        spl = line.split();
        u.append(float(spl[0]))
    x = linspace(0,1,len(u))
    plot(x,u,label=Label)
    title(plottitle)
    xlim(0,1); ylim(0,1)
    xlabel("x")
    ylabel("u")
    legend(loc="upper right")
    show()

def read_1D_movie(subdir,Title,plottitle,anititle):
    os.chdir('build-main-Desktop-Debug')

    if os.path.isdir(subdir):
        shutil.rmtree(subdir)
    os.mkdir(subdir)
    infile = open(Title)
    os.chdir(subdir)

    counter = 0
    for line in infile:
        splt = line.split()
        u = zeros(len(splt))
        for i in range(len(u)):
            u[i] = float(splt[i])

        x = linspace(0,1,len(u))

        plot(x,u,hold=False)
        title(plottitle)
        xlim(0,1); ylim(0,1)
        xlabel("x")
        ylabel("u(x)")
        savefig("tmp%04d.png"%counter)
        counter += 1

    os.system("mencoder 'mf://tmp*.png' -mf type=png:fps=10 -ovc lavc -lavcopts vcodec=wmv2 -oac copy -o %s"%anititle)
    infile.close()
    os.chdir(os.pardir)

#read_1D_MC_bin('build-main-Desktop-Debug/RandomWalk_a.txt',"Random walk with constant steplength. Plot by bin")
#read_1D_MC_bin('build-main-Desktop-Debug/RandomWalk_b.txt',"Random walk with constant steplength. Plot by bin")
#read_2D("build-main-Desktop-Debug/Explicit.txt","Explicit sheme for 2+1D diffusion equation")
#read_2D("build-main-Desktop-Debug/Implicit.txt","Implicit sheme for 2+1D diffusion equation")
#read_2D("build-main-Desktop-Debug/Analytic.txt","Closed form solution for 2D Laplace equation")
#read_1D('build-main-Desktop-Debug/Analytical_1d.txt',"Analytic solution to 1D diffusion equation","Analytic")
#read_1D('build-main-Desktop-Debug/CrankNicolson.txt',"Crank Nicolson solving 1D diffusion equation","CrankNicolson")
#read_1D_movie("analytical_plots", "Analytical_movie.txt", "Analytical solution to 1D diffusion","Analytic_1d.mpg")
#read_1D_movie("CrankNicolson_plots", "CrankNicolson_movie.txt", "Crank Nicolson, 1D diffusion","CrankNicolson.mpg")
#read_2D_movie("Implicit_movie.txt", "Implicit method", "Implicit_plots", "Implicit.mpg")
#read_2D_movie("Explicit_movie.txt", "Explicit method", "Explicit_plots", "Explicit.mpg")
read_1D_MC_movie("MC_uniform_movie.txt", "Uniform Random Walk", "Uniform_MC", "UniformMC.mpg")
#read_1D_MC_movie("MC_normal_movie.txt", "Gaussian Random Walk", "Normal_MC", "NormalMC.mpg")
```