```c
/*
 * ME 2054 Parallel Scientific Computing
 * Homework #2 - Couette Flow Solver
 * Due: October 1,2018
 *
 * Author: Dustin (Ting-Hsuan) Ma
 */
#include <stdlib.h>
#include <stdio.h>
#include <math.h>

int main(){

// Initializing variables
const float mu = 1.0, rho = 1.0, H = 2.0, numInt = 20.0;        // Arbitrary fluid values
float Re, PG, nu, Uplate, dy, dt, dp, nTimeStep, SteadyT;       // Internally defined values

//Allocating space for arrays
float *v = malloc(20*(sizeof *v));      // exact
float *u = malloc(20*(sizeof *u));      // numerical iteration
float *w = malloc(20*(sizeof *w));      // final numerical

// Asking for user input
printf("Please enter (1)Reynolds Number and (2)Pressure Gradient:\n");
scanf("%f%f", &Re, &PG);         // utilizing pointers
printf("You've chosen Re = %2.1f and P.G. = %1.1f\n",Re,PG);

// Calculation to define terms
dy = H / (numInt - 1);          // step size in y-axis
nu = mu / rho;                  // kinematic viscosity
Uplate = Re * nu / H;           // velocity of plate using user defined reynolds
dp = PG * (-rho);               // term for exact solution term
dt = 0.5 * (pow(dy,2) / nu);    // timestep
SteadyT = pow(H,2) / nu;        // expected steady state time
nTimeStep = SteadyT / dt;       // number of time step until steady state

// Calculating Exact Solution
int i = 0;
for (float y = 0; y < H + dy; y = y + dy){
        v[i] = Uplate * (y/H) + dp/(2*mu) * (pow(y,2) - H * y); // exact solution formula
        i++;    // incrementing index to store in array
}

// Calculating Numerical Solution
for (int i = 0; i < nTimeStep; i++){
        for (int j = 1; j < numInt; j++){
        u[j] = dt * (PG + mu * (w[j+1] - 2 * w[j] + w[j-1]) / pow(dy,2)) + w[j];

        // setting values into another vector for access in next iteration
        w[j] = u[j];

        // applying boundary conditions
        w[0] = 0;
        w[19] = Uplate;
}
}

// Output Final Solution
printf("Below shows the exact and numerical calculation for a couette flow.\n");
printf("You chose Re = %1.2f, and PG = %1.1f\n",Re,PG);
for (int i = 0; i < numInt; i++){
        printf("Exact = %-1.5f, Numerical = %-1.5f\n",v[i]/Uplate,w[i]/Uplate);
}

// Deallocating memory
free(v);free(w);free(u);
v = NULL;u=NULL;w=NULL;

return EXIT_SUCCESS;
}
```