"YOUR COMPANY NAME"

PUMP HEAD CALCULATION - CHILLED WATER - PRIMARY

HELP

* PRESSURE DROP BASED ON:
VELOCITY NOT EXCEEDING 5 FT/SEC. UP TO 2" PIPE
VELOCITY NOT EXCEEDING 10 FT/SEC. FOR LARGER THAN 2" PIPE
PRESSURE DROP BETWEEN 1 AND 5 FT/100 FT.

PROJECT NAME: TEST PROJECT #1
PROJECT NUMBER: 1444-03
DATE: 11/19/2003

APPROXIMATE PUMP BHP: 3.9 (CALCULATED DURING INPUT)

RESET/CLEAR

9 CAUTION : ABOVE OR BELOW NORMAL VELOCITY

ITEM	LENGTH (FT.)	SIZE (IN.)	NUMBER OF FITTINGS	AVERAGE GPM (EACH SIZE)	VELOCITY (FT./SEC.)	EQUIVALENT LENGTH (FT.)	PRESS. DROP -EACH (FT/100 FT.) *	LOSS FOR SIZE (FT. HD.)	TOTAL LOSS (FT. HD.)
PUMP ACCESSORIES:									
PUMP FLEX. CONNECTIONS		3	2	160	7	22	5.5	2.42	
CHECK VALVE - LIFT CHECK		3	1	160	7	84	5.5	4.62	
CHECK VALVE - SWING CHECK		0	0	0	0	0	0	0.00	
CHECK VALVE - SILENT CHECK		2	0	80	8	N/A	N/A	0.00	
BALANCING VALVE (80% OPEN)		3	1	160	7	N/A	N/A	4.00	
SHUT-OFF VALVE - BUTTERFLY		3	2	160	7	N/A	N/A	2.00	
SHUT-OFF VALVE - GATE		0	0	0	0	0	0	0.00	

?ref_=zg_vkf_q832_11 As far as I know, the loop that you have in your approach doesn't make sense - you could just move on with the 'calculate' function. But in order to handle collisions between rotated and non-rotated rectangles, you need a way to check if any two objects overlap. You have the y coordinate of each rectangle to work with, so you can use it to check if any two rectangles overlap. You can calculate the overlap between rectangles, then decide whether or not they should interact. An example that might help you: x1 y1 x2 y2 82157476af