```
syms E1 E2 E3;
maintenance = [(200+E2) (200+E2) (200+E2) (300+E2) (300+E2) (400+E2) (500+E2) (600+E2
    (50 +E3) (50 +E3) (100+E3) (150+E3) (150+E3) (200+E3) (250+E3) (300+E3) (350+E3)
E1=6;
E2=13;
E3=5;

line_optim_opt = optimoptions('linprog', 'Algorithm', 'dual-simplex');
quad_optim_opt = optimoptions('quadprog', 'Algorithm', 'interior-point-convex');

%data for Q3
measurement3  = readtable("measurements_physical.csv");

delta_t = 3600;
Qocc3   = measurement3{(1:2160),1};
Qac3    = measurement3{(1:2160),2};
Qvent3  = measurement3{(1:2160),3};
Qsolar3 = measurement3{(1:2160),4};
Tamb3   = measurement3{(1:2160),5};
TK1_3    = measurement3{(1:2160),6};
TK3     = measurement3{(1:2159),6};
Phi3    = measurement3{(1:2160),7};

%data for Q4
observations=readtable('measurements.csv')

q_dot_occ=table2array(observations([1:end-1],1));
q_dot_ac=table2array(observations([1:end-1],2));
q_dot_vent=table2array(observations([1:end-1],3));
q_dot_solar=table2array(observations([1:end-1],4));
T_amb=table2array(observations([1:end-1],5));
T_b=table2array(observations(:,6));


%% Q1_a & Q1_b

disp("Q1_a &Q1_b:")
c_1b = [-4 -2.5];
A_1b = [1 1;3000 1500];
b_1b = [12 24000 + 300*E1]';
lb_1b = [0 0]';
ub_1b = [inf inf]';
[x_1b,val_1b,flag_1b] = linprog(c_1b, A_1b, b_1b, [], [], lb_1b, ub_1b, [], line_optim
if flag_1b==1
    disp("installation plan");
    disp(x_1b);
    disp("maximum power");
    disp(-val_1b);
```

```
else
    disp("no optimal solutions");
end

disp("%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%");

%% Q1_c
disp("Q1_c");
maintenance=eval(maintenance);

acc_maintenance=cumsum(maintenance,2); %%maintenance accumulation by years
budget=4600*ones(1,10);
acc_budget=cumsum(budget); %%maintenance budget accumulation by years

x_1c=[];
val_1c=[];
flag_1c=[];
f_1c=-[4,2.5];

for i=1:1:10
    A_1c=[1,1;3000+acc_maintenance(1,i),1500+acc_maintenance(2,i)];
    b_1c=[12,25800+acc_budget(1,i)];
    lb_1c=[0,0]';
    ub_1c=[];
    [tem_x,tem_val,tem_flag]=linprog(f_1c,A_1c,b_1c,[],[],lb_1c,ub_1c,line_optim_opt)
    x_1c=[x_1c,tem_x];
    val_1c=[val_1c,tem_val];
    flag_1c=[flag_1c,tem_flag];
end

disp('installation plan for different duration:');
disp(x_1c);
disp('maximum power for different duration');
disp(-val_1c);
disp('optimization flag for different duration');
disp(flag_1c);

duration=max(-val_1c);

disp("without considering the answer should be integral:")
[opt_val,duration_time]=max(-val_1c);
max_power_1c=x_1c(:,duration_time);
disp("the best durable years");
disp(duration_time);
disp("best installation plan");
disp(max_power_1c);
disp('maximum power');
disp(max(-val_1c))
```

```
%integral answer for duration time 5
possi_instal_plans=[10,10,11;1,2,1]; %possible integer solutions for durable years 5
A_cost_1c=[3000+acc_maintenance(1,5),1500+acc_maintenance(2,5)];
b_cost_1c=25800+acc_budget(1,5);
possi_cost=zeros(1,3);
possi_power=zeros(1,3);

for i=1:1:3
    possi_cost(i)=A_cost_1c*possi_instal_plans(:,i);
    possi_power(i)=[4,2.5]*possi_instal_plans(:,i);
    if possi_cost(i)>b_cost_1c %whether meet constraints or not
        possi_cost(i)=-1;
        possi_power(i)=-1;
    end
end

[max_power_1c,pos]=max(possi_power);
instal_plan_1c=possi_instal_plans(:,pos);

disp("considering integer constraints, for 5 years:");
disp("optimal integer installation plan:");
disp(instal_plan_1c);
disp("optimal power:");
disp(max_power_1c);
disp("However, if duration time is chosen to be 4,6,7"+...
    " it seems also possible to obtain this plan");

%test other duration years
possi_cost=zeros(4,3);
possi_power=zeros(1,3);
for i=1:1:4
    A_cost_1c=[3000+acc_maintenance(1,i+3),1500+acc_maintenance(2,i+3)];
    b_cost_1c=25800+acc_budget(1,i+3);
    for j=1:1:3
        possi_cost(i,j)=A_cost_1c*possi_instal_plans(:,j);
        possi_power(i,j)=[4,2.5]*possi_instal_plans(:,j);
        if possi_cost(i,j)>b_cost_1c
            possi_cost(i,j)=-1;
            possi_power(i,j)=-1;
        end
    end
end

disp("check all possible install plan for duration time 4,5,6,7");
disp(possi_instal_plans);
disp("check all possible maximum power for duration time 4,5,6,7");
disp(possi_power);
disp("And now we can find, for duration time 4,5,6,7, choose 10 x_1b and 2 Y "+...
    "we can obtain optimal power 45kW");
```

```
disp('%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%');

%% Q3
disp("Q3:");
%task3
Qsigma3 = Qocc3(1:2159,1) - Qvent3(1:2159,1) + Qac3(1:2159,1);
Tsigma3 = Tamb3(1:2159,1) - TK3(1:2159,1);

phi_3 = delta_t*[Qsolar3(1:2159,1) Qsigma3 Tsigma3];
Y_3 = TK1_3(2:2160,1) - TK3(1:2159,1);
H_3 = 2*(phi_3') * phi_3;
C_3 = -2*(Y_3') * phi_3;
% Five = [TK Q];
% H2 = Five' * Five;
% c2 = -2 * TK1' * Five;
% lb_3 = [-Inf -Inf];
% ub_3 = [Inf Inf];
[x_3,val_3,flag_3] = quadprog(H_3, C_3, [], [], [], [], [], [], [], quad_optim_opt);

if flag_3==1
    disp("value of a1,a2,a3 are:");
    disp(x_3);
else
    disp("no optimal solutions");
end
disp('%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%');

%% Q_4
disp("Q4")

N=2160;
T1=22.43;
additional_cost=0.1+13/10;
price_phi=table2array(observations(:,end));

T_min=ones(N-1,1)*15;
T_max=ones(N-1,1)*28;
T_ref=ones(N-1,1)*22;
q_dot_ac_max=ones(N,1)*100;
lb_4=[zeros(N,1);T_min];
ub_4=[q_dot_ac_max;T_max];

for i=1:1:N-1
    if (q_dot_occ(i)<0)
        lb_4(N+i-1)=-inf;
        ub_4(N+i-1)=inf;
    end
end
```

```
c_4=[price_phi;-2*additional_cost*T_ref];

H_4=zeros(2160*2-1,2160*2-1);
for i=2161:2*2160-1
    H_4(i,i)=additional_cost*2;
end

parameter_A=1-x_3(3)*delta_t;

parameter_B=[x_3(1)*delta_t,x_3(2)*delta_t,x_3(2)*delta_t,...
    -x_3(2)*delta_t,x_3(3)*delta_t];

beq_4=parameter_B(1)*q_dot_solar+parameter_B(2)*q_dot_occ+...
    parameter_B(4)*q_dot_vent+parameter_B(5)*T_amb;
beq_4(1)=beq_4(1)+T1*parameter_A;

A1=diag(-parameter_B(3)*ones(1,N-1));
A2=zeros(N-1,1);
A3=diag(ones(1,N-1))+diag(-ones(1,N-1-1)*parameter_A,-1);
Aeq_4=[A1,A2,A3];

[x_4,fval_4,flag_4]=quadprog(H_4,c_4,[],[],Aeq_4,beq_4,lb_4,ub_4,[],quad_optim_opt);

if flag_4==1
    fval_4=fval_4+1.4*2160*22^2+1.4*T1^2-2.8*22*T1;
    disp("the optimal cost for air-conditioning along the horizon of N steps");
    disp(fval_4);
else
    disp("no optimal solutions");
end
```