function main\_2()

% Initialize variables

C = 0;

totalTime = 0;

maxOutput = 0;

optimalPath = [];

RGVPosition = 1; % Assuming RGV starts at position 1

CNCProcessingTime = zeros(1, 8); % Time required for each CNC to process a workpiece

CNCWorkStatus = zeros(1, 8); % 1 if CNC is processing, 0 if not

%materialDemandPoints = [1,2,3,4,5,6,7,8]; % List of material demand points (CNC machine locations)

upload1 = [];

upload2 = [];

download1 = [];

download2 = [];

k = 0;

Tz = [27,32];%[28,31];[30,35];[27,32];

Tq = 25;%25;30;25;

while totalTime <= 28800

k = k + 1;

% Step 1: Calculate the shortest time required to move to the next material demand point

[shortestTime,nextCNC] = calculateShortestTime(C, k, RGVPosition, CNCWorkStatus, CNCProcessingTime);

% Step 2: Move RGV to the next material demand point

moveTime = shortestTime;

totalTime = totalTime + moveTime;

RGVPosition = nextCNC;

[CNCProcessingTime, CNCWorkStatus] = Elapsing(CNCProcessingTime, CNCWorkStatus, moveTime);

if C == 0, upload1 = [upload1; totalTime];end

if C == 1, upload2 = [upload2; totalTime];end

if C == 0 && k >=6, download1 = [download1; totalTime]; end

if C == 1 && k >=11, download2 = [download2; totalTime]; end

% Step 3: Perform RGV material loading/unloading operations and update CNC status

totalTime = totalTime + Tz(2-mod(RGVPosition,2));

[CNCProcessingTime, CNCWorkStatus] = Elapsing(CNCProcessingTime, CNCWorkStatus, Tz(2-mod(RGVPosition,2)));

CNCProcessingTime(RGVPosition) = eps;

CNCWorkStatus(RGVPosition) = 1;

if k >= 6, C = 1 - C; end

% Step 4: Clean the material and complete a cycle

optimalPath = [optimalPath; RGVPosition];

if C == 1 && k >= 11

maxOutput = maxOutput + 1;

totalTime = totalTime + Tq;

[CNCProcessingTime, CNCWorkStatus] = Elapsing(CNCProcessingTime, CNCWorkStatus, Tq);

end

% Step 5: Update total time and check if it exceeds 8 hours

if totalTime <= 28800

% Update the max output and optimal path if needed

else

break; % Exit the loop if the total time exceeds 8 hours

end

end

% Output the results

fprintf('Maximum material output: %d\n', maxOutput);

fprintf('totalTime: %d\n', totalTime);

fprintf('Optimal path: %s\n', mat2str(optimalPath));

fprintf('upload1: %s\n', mat2str(upload1));

fprintf('download1: %s\n', mat2str(download1));

fprintf('upload2: %s\n', mat2str(upload2));

fprintf('download2: %s\n', mat2str(download2));

end

function [shortestTime, nextCNC] = calculateShortestTime(C, k, RGVPosition, CNCWorkStatus, CNCProcessingTime)

Tj1 = 455;%400;280;455;

Tj2 = 182;%378;500;182;

I = [1,2,3,5,7];

J = [4,6,8];

Time1 = zeros();

Time2 = zeros();

Time = [inf,inf,inf,inf,inf,inf,inf,inf];

if k == 1, nextCNC = 1; shortestTime = calculateMovingTime(RGVPosition,nextCNC);end

if k == 2, nextCNC = 2; shortestTime = calculateMovingTime(RGVPosition,nextCNC);end

if k == 3, nextCNC = 3; shortestTime = calculateMovingTime(RGVPosition,nextCNC);end

if k == 4, nextCNC = 7; shortestTime = calculateMovingTime(RGVPosition,nextCNC);end

if k == 5, nextCNC = 5; shortestTime = calculateMovingTime(RGVPosition,nextCNC);end

if k == 6, nextCNC = 1; shortestTime = Tj1-CNCProcessingTime(1);end

if k > 6

if C == 0

for i = 1:5

if CNCWorkStatus(I(i)) == 0

Time(I(i)) = calculateMovingTime(RGVPosition,I(i));

else

Time1(I(i)) = calculateMovingTime(RGVPosition,I(i));

Time2(I(i)) = Tj1-CNCProcessingTime(I(i));

Time(I(i)) = max(Time1(I(i)), Time2(I(i)));

end

end

[shortestTime, nextCNC] = min(Time);

else

for j = 1:3

if CNCWorkStatus(J(j)) == 0

Time(J(j)) = calculateMovingTime(RGVPosition,J(j));

else

Time1(J(j)) = calculateMovingTime(RGVPosition,J(j));

Time2(J(j)) = Tj2-CNCProcessingTime(J(j));

Time(J(j)) = max(Time1(J(j)), Time2(J(j)));

end

end

[shortestTime, nextCNC] = min(Time);

end

end

end

function [MovingTime] = calculateMovingTime(RGVPosition,n)

if(abs(((RGVPosition+mod(RGVPosition,2))/2-1)-((n+mod(n,2))/2-1))==0)

MovingTime=0;

elseif(abs(((RGVPosition+mod(RGVPosition,2))/2-1)-((n+mod(n,2))/2-1))==1)

MovingTime=18;%20;23;18;

elseif(abs(((RGVPosition+mod(RGVPosition,2))/2-1)-((n+mod(n,2))/2-1))==2)

MovingTime=32;%33;41;32;

else,MovingTime=46;%46;59;46;

end

end

function [CNCProcessingTime, CNCWorkStatus] = Elapsing(CNCProcessingTime, CNCWorkStatus, T)

Tj1 = 455;%400;280;455;

Tj2 = 182;%378;500;182;

CNCProcessingTime = CNCProcessingTime + T.\* CNCWorkStatus;

for i = 1:8

if i == 1||2||3||5||7

if CNCProcessingTime(i) >= Tj1

CNCProcessingTime(i) = 0;

CNCWorkStatus(i) = 0;

end

else

if CNCProcessingTime(i) >= Tj2

CNCProcessingTime(i) = 0;

CNCWorkStatus(i) = 0;

end

end

end

end