//PreemptiveSJF

import java.util.Scanner;

public class PreemptiveSJF {

    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);

        System.out.println("enter no of process: ");

        int n = sc.nextInt();

        int pid[] = new int[n];   // process ids

        int ar[] = new int[n];     // arrival times

        int bt[] = new int[n];     // burst or execution times

        int ct[] = new int[n];     // completion times

        int ta[] = new int[n];     // turn around times

        int wt[] = new int[n];     // waiting times

                int f[] = new int[n];

                int k[] = new int[n];

        int temp, tot=0,st=0;

        float avgwt=0,avgta=0;

        for(int i = 0; i < n; i++)

        {

            System.out.println("enter process " + (i+1) + " arrival time: ");

            ar[i] = sc.nextInt();

            System.out.println("enter process " + (i+1) + " brust time: ");

            bt[i] = sc.nextInt();

                                                    k[i] = bt[i];

            pid[i] = i+1;

        }

                                    while(true){

                                        int min = 99 , c =n;

                                            if(tot == n)

                                                    break;

                                            for(int i=0;i<n;i++){

                                                if(ar[i]<=st && f[i] == 0 && bt[i]<min ){

                                                    min = bt[i];

                                                    c = i;

                                                }

                                            }

                                            if (c == n) st++;

                                            else{

                                                bt[c]--;

                                                st++;

                                                if(bt[c] == 0){

                                                    ct[c] =st;

                                                    f[c]=1;

                                                    tot++;

                                                }

                                            }

                                    }

                                    for(int i=0;i<n;i++){

                                        ta[i] = ct[i] - ar[i];

                                        wt[i] = ta[i] - k[i];

                                        avgwt += wt[i];

                                        avgta += ta[i];

                                    }

                                    System.out.println("pid  arrival  burst  complete turn waiting");

        for(int i=0;i<n;i++)

        {

            System.out.println(pid[i] +"\t"+ ar[i]+"\t"+ k[i] +"\t"+ ct[i] +"\t"+ ta[i] +"\t"+ wt[i]);

        }

        System.out.println("\naverage tat is "+ (float)(avgta/n));

        System.out.println("average wt is "+ (float)(avgwt/n));

        sc.close();

    }

}