```
#include<stdio.h>
#include<stdlib.h>


struct process{
   int pid;
   int arrival_time;
   int burst_time;
   int FinishTime;
   int TurnaroundTime;
   int WaitingTime;
};


void print_struc(struct process proc[],int p){
   printf("Process\tAT\tBT\tFT\tTT\tWT\n");
   for(int i=0;i<p;i++){

printf("%d\t%d\t%d\t%d\t%d\t%d\n",proc[i].pid,proc[i].arrival_time,proc[i].burst_time,proc[i].Finish
Time,proc[i].TurnaroundTime,proc[i].WaitingTime);
   }
}


int main(){

   int p;
   printf("Enter the no. of processes:");
   scanf("%d",&p);
   struct process proc[p];
   for(int i=0;i<p;i++){
      proc[i].pid=i;
   }
   printf("Enter the arrival time for the processes:");
   for(int i=0;i<p;i++){
      scanf("%d",&proc[i].arrival_time);
   }
   printf("Enter the burst time for the processes:");
   for(int i=0;i<p;i++){
      scanf("%d",&proc[i].burst_time);
   }
   for(int i =0;i<p-1;i++){
      for(int j=0;j<p-i-1;j++){
         if(proc[j].arrival_time>proc[j+1].arrival_time){
            struct process temp = proc[j];
            proc[j]=proc[j+1];
            proc[j+1]=temp;
         }
      }
   }
   int curr_finish=proc[0].arrival_time;
   for(int i=0;i<p;i++){
      proc[i].FinishTime=proc[i].burst_time+curr_finish;
```

```
      curr_finish=proc[i].FinishTime;
   }
   for(int i=0;i<p;i++){
      proc[i].TurnaroundTime=proc[i].FinishTime-proc[i].arrival_time;
   }
   for(int i=0;i<p;i++){
      proc[i].WaitingTime=proc[i].TurnaroundTime-proc[i].burst_time;
   }

   print_struc(proc,p);


   printf("Following is the gantt chart\n");
   for(int i=0;i<p;i++){
      printf("%d\t\t",proc[i].pid);
   }printf("\n");
   for(int i=0;i<p;i++){
      printf("%d\t\t",proc[i].FinishTime);
   }







   return 0;
}



ALTERNATE  CODE -->


#include<stdio.h>
#include<stdlib.h>

// FOR FCFS---->>>>

int main()
{// FOR FCFS---->>>>
   int n;
   printf("Enter the number of processes:");
   scanf("%d", &n);

   int burst_time[n];
   int arrival_time[n];
   int comp_time[n];
   int tat[n];
   int wait[n];
   int comp[n];
   printf("Enter the arrival time for each process:\n");

   for (int i = 0; i < n; i++)
   {
      printf("arrival time of process %d: ", i + 1);
      scanf("%d", &arrival_time[i]);
```

```
    }

    printf("Enter the burst time for each process:\n");

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

        printf("Burst time of process %d: ", i + 1);

        scanf("%d", &burst_time[i]);
    }


    for(int i=0;i<n;i++){
        comp_time[i]=0;
        for(int j=i;j>=0;j--){
            comp_time[i]=comp_time[i]+burst_time[j];
            comp[i]=comp_time[i];
        }
    }
    for(int i=0;i<n;i++){

        tat[i]=comp[i]-arrival_time[i];
    }
    printf("completion time of processes are :");
    for(int i=0;i<n;i++){

        printf("%d--",i+1);
        printf("%d ",comp[i]);
    }
    printf("tat time of processes : ");
    for(int i=0;i<n;i++){

        printf("%d--",i+1);
        printf("%d ",tat[i]);
    }




    return 0;
}
```