APPENDIX

PSEUDO CODE FOR JOB SCHEDULING

Prompt the value from the user
BEGIN
READ Job name, Arrival time and burst time
ser1[i] = st[i]; x[i] = 2;
BEGIN IF(i==0) THEN
    y[i] = 4;
ELSE
    y[i] = y[i-1]+6;
    wt[i] = 0;
    wait[i] = 0;
    status[i] =0;
    jend[i]=0;
    ttpt += st[i];
PRINT Service time, waiting time, Turnaround time, Response time
END

FCFS

FOR 1 TO N
BEGIN IF(i==0) THEN
    wt[i] = 0;
ELSE
    wt[i] = wt[i-1] + b[i-1];
    start = clock();
    jobs(jname[i],b[i],wt[i]);
    ttpt += wt[i-1];
    tp = (float)n / (float) ttpt;
    EFFICIENCY=I/tp;
END FOR
END IF
PRINT Total throughput time, Efficency, throughput
END

SJFS

FOR 1 TO N
BEGIN
IF(i==0)THEN
    wt[i] = 0;
ELSE
    wt[i] = wt[i-1] + b[i-1];
    start = clock();
    jobs(jname[i],b[i],wt[i]);
    ttpt += wt[i-1];
    tp = (float)n / (float) ttpt;
    EFFICIENCY=1/tp;
END FOR
END IF
PRINT Total throughput time, Efficency, throughput
END

PRIORITY

FOR 1 TO N
BEGIN
    READ P[i]
    IF(i==0)
        wt[i] = 0;
    ELSE
        wt[i] = wt[i-1] + b[i-1];
        start = clock();
        jobs(jname[i],b[i],wt[i]);
        ttpt += wt[i-1];
    END IF
END FOR
PRINT Total throughput time, Efficency, throughput
END
tp = (float)n / (float) tpt;
EFFICIENCY = 1/tp;
END FOR
END IF
PRINT Total throughput time, Efficiency, throughput
END

ROUND ROBIN

READ Quantum time
BEGIN
texattr(14);
i=0;
start1 = clock();
END
BEGIN DO-WHILE
BEGIN
IF(st[i] !=0) THEN
FOR 1 TO N
IF (i!=j) THEN
IF(jend[j]==0) THEN
wt[i] += wait[j];
job();
end1 = clock();
wait[i] = (end1 - start)/CLK_TCK;
ELSE
jend[i]++;
i++;
IF (i==n) THEN
i=0;
WHILE(core());
END WHILE
END WHILE
FOR I 1 TO N
twt += wt[i];
tpt += twt;
tp = (float)n / (float)tpt;
EFFICIENCY=1/tp;
END FOR
END IF
    PRINT  Total throughput time, Efficiency, throughput
END