OPERATING SYSTEMS

CSE 2005

L23+L24

SURVEY OF EFFICIENCY OF THE CPU SCHEDULING ALGORITHM

UNDER THE GUIDANCE OF

PROF. SELVAKUMAR K

TANISHQ HITESH BAFNA -17BCE2001

KARNA GOPICHAND REDDY :-17BCE0244

CHALLA SAI YASHWATH:-17BCE0827

ABSTRACT

DEVELOPING CPU SCHEDULING ALGORITHMS AND UNDERSTANDING THEIR IMPACT IN PRACTICE CAN BE DIFFICULT AND TIME CONSUMING DUE TO THE NEED TO MODIFY AND TEST OPERATING SYSTEM KERNEL CODE AND MEASURE THE RESULTING PERFORMANCE ON A CONSISTENT WORKLOAD OF REAL APPLICATIONS. AS PROCESSOR IS THE IMPORTANT RESOURCE, CPU SCHEDULING BECOMES VERY IMPORTANT IN ACCOMPLISHING THE OPERATING SYSTEM (OS) DESIGN GOALS. THE INTENTION SHOULD BE ALLOWED AS MANY AS POSSIBLE RUNNING PROCESSES AT ALL TIME IN ORDER TO MAKE BEST USE OF CPU. THIS PAPER PRESENTS A STATE DIAGRAM THAT DEPICTS THE COMPARATIVE STUDY OF VARIOUS SCHEDULING ALGORITHMS FOR A SINGLE CPU AND SHOWS WHICH ALGORITHM IS BEST FOR THE PARTICULAR SITUATION. USING THIS REPRESENTATION, IT BECOMES MUCH EASIER TO UNDERSTAND WHAT IS GOING ON INSIDE THE SYSTEM AND WHY A DIFFERENT SET OF PROCESSES IS A CANDIDATE FOR THE ALLOCATION OF THE CPU AT DIFFERENT TIME. THE OBJECTIVE OF THE STUDY IS TO ANALYZE THE HIGH EFFICIENT CPU SCHEDULER ON DESIGN OF THE HIGH QUALITY SCHEDULING ALGORITHMS WHICH SUITS THE SCHEDULING GOALS. KEY WORDS:-SCHEDULER, STATE DIAGRAMS, CPU-SCHEDULING, PERFORMANCE .

KEYWORDS

MULTIPROGRAMMING, SCHEDULING, PROCESS, FCFS, SJF,PRIORITY, ROUND ROBIN .

INTRODUCTION

THERE ARE DIFFERENT TYPES OF CPU SCHEDULING ALGORITHM SUCH AS FCFS(FIRST COME FIRST SERVED), SJF(SHORTEST JOB FIRST) PRE-EMPTIVE / NON PRE-EMPTIVE, PRIORITY , ROUND ROBIN ,MULTILEVEL QUEUE. BUT THERE IS A QUESTION THAT WHICH ALGORITHM TO USED .WE HAVE TO TAKE CARE OF THE EFFICIENCY OF THE ALGORITHM. IN THIS REPORT WE ARE GOING TO DO SURVEY OF WHICH ALGORITH IS THE MOST EFFICIENT . WE HAVE TO CONSIDER THE AVERAGE WAITING AND AVERAGE TURN AROUND TIME. THERE MAY BE DIFFERENT COMBINATION OF PROCESS WITH DIFFERENT BURST TIME / PRIORITY / ETC. WE WILL HAVE COMBINATION OF TWO ALGORITHM SUCH AS PRIORITY WITH ROUND ROBIN PRE-EMPTIVE AND NON PRE-EMPTIVE. AFTER ALL THE CALCULATION WITH THE HELP OF THE ABOVE ALGORITHM WE WILL FIND THE

METHODOLOGY

IN THIS PROJECT WE ARE GOING TO TRY TO USE A NEW TYPE OF CPU SCHEDULING WHICH WOULD BE A COMBINATION OF SHORTEST JOB FIRST AND ROUND ROBIN. AS SJF HAS MAJORITY OF THE TIME BEEN MORE EFFICIENT THUS TRYING TO ADD ROUND ROBIN WITH IT WOULD ENHANCE ROUND ROBIN.

WE WILL BE FINDING TIME QUANTUM USING DYNAMIC TIME QUANTUM BY THE MATHEMATICAL FORMULA

TQ=MAX((MEDIAN+HIGHEST+LOWEST)/2)

RELATED WORKS

PROBLEM 1

PROCESS	BURST TIME	ARRIVAL TIME	PRIORITY
1	10	3	2
2	15	1	3
3	8	6	1
4	3	0	4
5	6	10	5

FCFS

P4	P2	P1	P3	P5

0 3 18 28 36 42

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	15	10	25
P2	2	15	17
P3	22	8	30
P4	0	3	3
P5	26	6	32

AVERAGE WAITING TIME=65/5=13MS

AVERAGE TURNAROUND TIME=107/5=21.4MS

SJF

P4	P5	P3	P1	P2

0 3 9 17 27 42

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	17	10	27
P2	27	15	32
P3	9	8	17
P4	0	3	3
P5	3	6	9

AVERAGE WAITING TIME=56/5=11.2MS

AVERAGE TURNAROUND TIME=98/5=19.6MS

SRTF

P4	P1	P1	P5	P3	P2

0 3 6 13 19 27 42

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	0	10	10
P2	26	15	41
P3	13	8	21
P4	0	3	3
P5	0	6	6

AVERAGE WAITING TIME=39/5=7.8MS

AVERAGE TURNAROUND TIME=81/5=16.2MS

PRIORITY

P3	P1	P2	P4	P5

0 8 18 33 36 42

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	8	10	18
P2	18	15	33
P3	0	8	8
P4	33	3	36
P5	36	6	42

AVERAGE WAITING TIME=95/5=19MS

AVERAGE TURNAROUND TIME=137/5=27.4MS

PRIORITY (PRE-EMPTIVE)

P4	P2	P1	P3	P3	P1	P2	P4	P5	
0	1	3	6	10	14	21	34	36	42

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	8	10	18
P2	18	15	33
P3	0	8	8
P4	33	3	36
P5	36	6	42

AVERAGE WAITING TIME=95/5=19MS

AVERAGE TURNAROUND TIME=127/5=25.4MS

PROBLEM 2

PROCESS	ARRIVAL TIME	BURST TIME	PRIORITY
P1	0	3	10
P2	1	2	6
P3	3	2	4
P4	5	4	3
P5	6	1	7
P6	7	5	11
P7	9	6	8
P8	12	4	9
P9	13	3	1
P10	15	2	5
P11	15	5	2
P12	15	6	13
P13	18	7	12

FCFS

P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13

0 3 5 7 11 12 17 23 27 30 32 37 43 50

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	0	3	3
P2	2	2	4
P3	2	2	4
P4	2	4	6
P5	5	1	6
P6	5	5	10
P7	8	6	14
P8	11	4	15
P9	14	3	17
P10	15	2	17
P11	17	5	22
P12	22	6	28
P13	25	7	32

AVERAGE WAITING TIME=128/13=9.846MS

AVERAGE TURNAROUND TIME=178/13=13.692MS

SJF(PREEMPTIVE)

P1	P1	P2	P3	P3	P5	P4	P4	P8	P8

0 1 3 5 6 7 8 9 12 13 15

P8	P10	P9	P6	P11	P7	P12	P13

15 16 18 21 26 31 37 43 50

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	0	3	3
P2	2	2	4
P3	2	2	4
P4	3	4	7
P5	1	1	2
P6	14	5	19
P7	22	6	28
P8	0	4	4
P9	5	3	8
P10	1	2	3
P11	11	5	16
P12	22	6	28
P13	25	7	32

AVERAGE WAITING TIME=108/13=8.307MS

AVERAGE TURNAROUND TIME=158/13=12.153MS

SJF(NON PREEMPTIVE)

P1	P2	P3	P5	P4	P8	P10	P9	P6	P11

0 3 5 7 8 12 16 18 21 26 31

P7	P12	P13

31 37 43 50

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	0	3	3
P2	2	2	4
P3	2	2	4
P4	3	4	7
P5	1	1	2
P6	14	5	19
P7	22	6	28
P8	0	4	4
P9	5	3	8
P10	1	2	3
P11	11	5	17
P12	22	6	28
P13	25	7	32

AVERAGE WAITING TIME=108/13=8.307MS

AVERAGE TURNAROUND TIME=158/13=12.153MS

PRIORITY(PREEMPTIVE)

P1	P2	P3	P4	P4	P4	P5	P7	P7	P9

0 1 3 5 6 7 9 10 12 13 15

P9	P11	P11	P10	P7	P1	P8	P6	P13	P12

15 16 18 21 23 26 28 32 37 44 50

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	25	3	28
P2	0	2	2
P3	0	2	2
P4	0	4	4
P5	3	1	4
P6	25	5	30
P7	11	6	17
P8	16	4	20
P9	0	3	3
P10	6	2	8
P11	1	5	6
P12	29	6	35
P13	19	7	26

AVERAGE WAITING TIME=135/13 =10.384MS

AVERAGE TURNAROUND TIME=185/13 =14.230MS

PRIORITY(NON PREEMPTIVE)

P1	P3	P4	P2	P5	P7	P9	P11	P10	P8	
0	3	5	9	11	12	18	21	26	28	32

P6	P13	P12	
32	37	44	50

PROCESS	WAITING TIME	BURST TIME	TURNAROUND TIME
P1	0	3	3
P2	8	2	10
P3	0	2	2
P4	0	4	4
P5	5	1	6
P6	25	5	30
P7	3	6	9
P8	16	4	20
P9	5	3	8
P10	11	2	13
P11	6	5	11
P12	29	6	35
P13	19	7	26

AVERAGE WAITING TIME=127/13=9.769MS

AVERAGE TURNAROUND TIME=177/13=13.615MS

PROPOSED METHOD/WORKING METHODOLOGY/ALGORITHMS

SJF+ROUND ROBIN

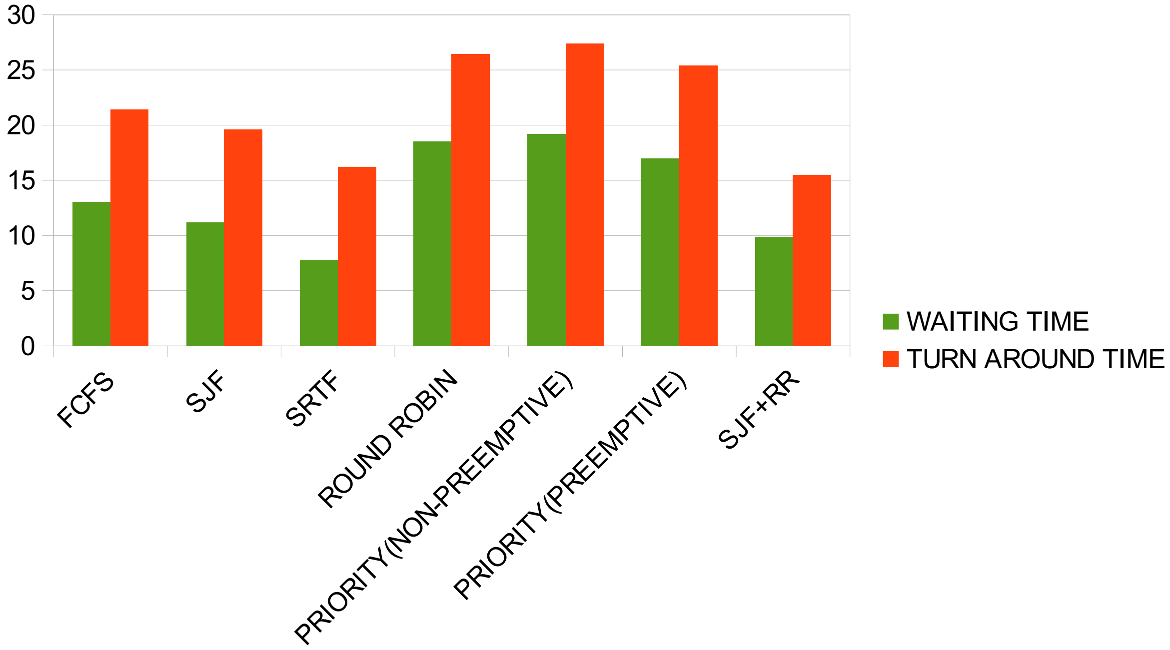
```
INT GET_TQ(INT B[],INT S)
{
INT I,J,MAXBT,TMP,HBT,MEDIAN;
FLOAT K,L,M;
FOR(I=0;I<S;I++)
{
        FOR(J=I+1;J<S;J++)
        {
                IF (B[I]>B[J])
                {
                TMP=B[I]; B[I]=B[J]; B[J]=TMP;
                }
        }
}
HBT=B[I-1];
MEDIAN=B[I/2];
FOR(I=0;I<S;I++)
ST[I]=B[I];
L=(FLOAT)HBT;
M=(FLOAT)MEDIAN;
K=SQRT((L*M))/2;
 RETURN(CEIL(K));
}

INT MAIN()
  {
  INT BT[10],WT[10],TAT[10],N,TQ;
   INT I,COUNT=0,SWT=0,STAT=0,TEMP,SQ=0;
        FLOAT AWT=0.0,ATAT=0.0;
        PRINTF("ENTER NUMBER OF PROCESSES:");
        SCANF("%D",&N); PRINTF("ENTER BURST TIME FOR SEQUENCES:");
        FOR(I=0;I<N;I++)
        {
        SCANF("%D",&BT[I]); ST[I]=BT[I];
        }
        TQ=GET_TQ(ST,N);
        PRINTF("\NTIME QUANTUM IS COMPUTED BY CEIL((HIGHESTBT+MEDIAN)/2) = %D\
N",TQ);
        WHILE(1)
        {
         FOR(I=0,COUNT=0;I<N;I++)
         {
```

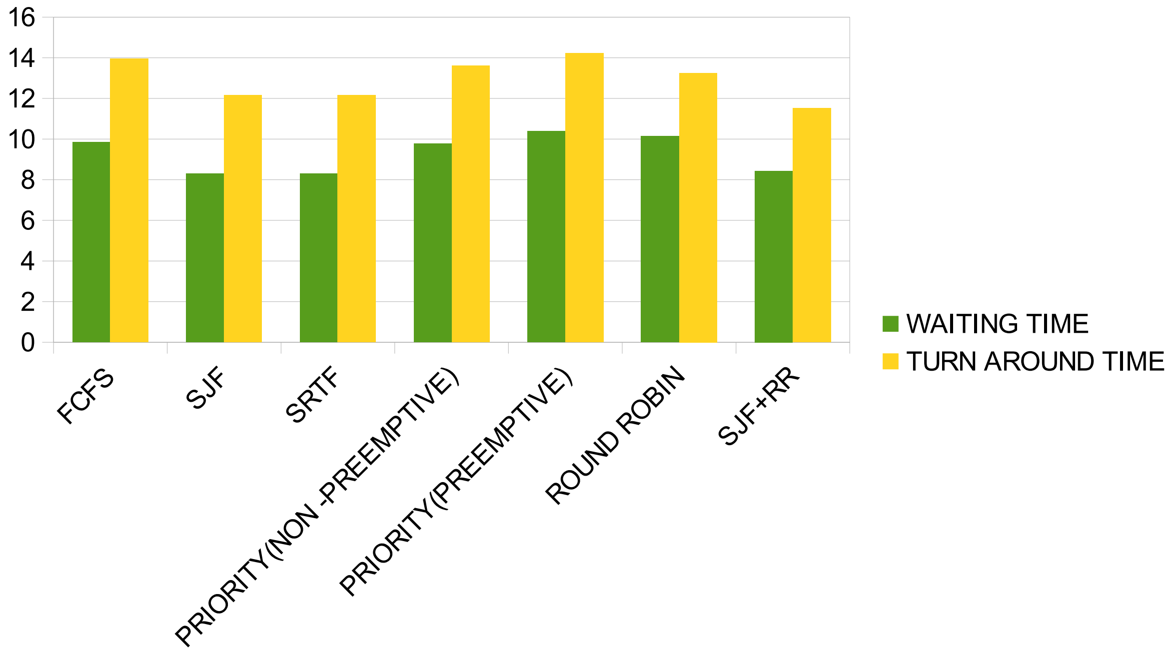
```
TEMP=TQ;
 IF(ST[I]==0)
 {
  COUNT++; CONTINUE;
 }
 IF(ST[I]>TQ)
            ST[I]=ST[I]-TQ;
            ELSE IF(ST[I]>=0)
            {
            TEMP=ST[I];
            ST[I]=0;
            }
            SQ=SQ+TEMP;
            TAT[I]=SQ;
            }
            IF(N==COUNT) BREAK;
}
            FOR(I=0;I<N;I++)
            {
             WT[I]=TAT[I]-BT[I];
             SWT=SWT+WT[I];
             STAT=STAT+TAT[I];
            }
             AWT=(FLOAT)SWT/(1.5*N);
             ATAT=(FLOAT)STAT/(1.5*N);
PRINTF("\NAVG WAITING TIME IS %F\NAVG TURN AROUND TIME IS %F",AWT,ATAT);
            GETCH();
}
```

RESULT & DISCUSSION

PROBLEM 1

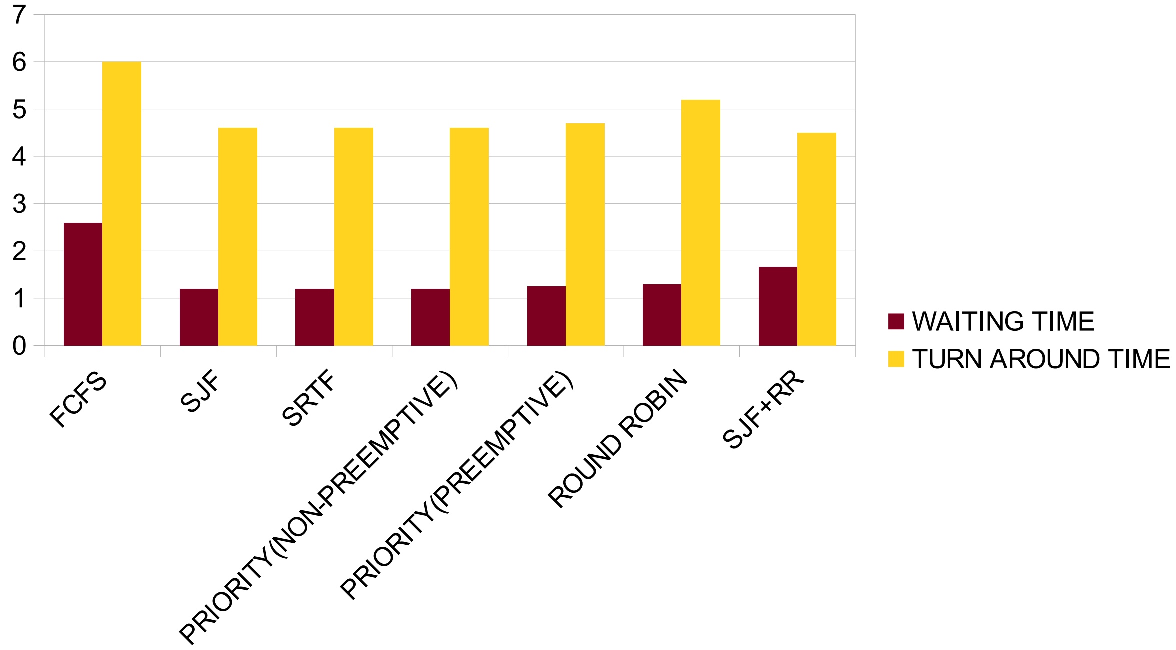


PROBLEM 2



PROBLEM 3

PROCESS	BURST TIME	WAITING TIME	PRIORITY
1	8	5	3
2	1	7	1
3	3	0	3
4	1	2	4
5	4	4	2



FROM THE ABOVE RESULT WE CAN SEE THAT SHORTEST REMAINING TIME FIRST (SJF PRE-EMPTIVE) AND SJF+ROUND ROBIN HAVE PROVED TO BE THE, MOST EFFICIENT CPU SCHEDULING ALGORITHM FOR SJF +ROUND ROBIN WE USE DYNAMIC TIME QUANTUM TECHNIQUE TO FIND THE TIME QUANTUM.

CONCLUSION

IN THIS PROJECT AN IMPROVEMENT FOR THE CONVENTIONAL ROUND ROBIN ALGORITHM IS PROPOSED WHICH IS BEING SUPPORTED BY A SET OF HYPOTHETICAL EXAMPLES AND A BETTER AMOUNT OF IMPROVEMENT IS OBSERVED. THE APPROACH CAN BE FURTHER REFINED USING THE CONCEPT OF ARRIVAL TIME.

REFERENCES

[1] SILBERSCHATZ, A., PETERSON, J. L., AND GALVIN, P.B., OPERATING SYSTEM CONCEPTS, ADDISON WESLEY, 7TH EDITION, 2006.

[2] STALLINGS, W., “OPERATING SYSTEMS: INTERNALS AND DESIGN PRINCIPLES”, 3RD EDT., PRENTICE HALL, INC., 1998, PP. 394

[3] ANDREW S. TANENBAUM , AND ALBERT S. WOODFHULL , OPERATING SYSTEMS DESIGN AND IMPLEMENTATION, SECOND EDITION, 2005. 17

[4] JEFFAY, K.; DONELSON SMITH, F.; MOORTHY, A.; ANDERSON, J.;” PROPORTIONAL SHARE SCHEDULING OF OPERATING SYSTEM SERVICES FOR REAL-TIME APPLICATIONS”, REAL-TIME SYSTEMS SYMPOSIUM, 1998. PROCEEDINGS. THE 19TH IEEE, 2-4 DEC. 1998 PAGE(S):480 – 491

[5]. NOON A, KALAKECH A, KADRY S.A NEW ROUND ROBIN BASED SCHEDULING ALGORITHM FOR OPERATINGSYSTEMS: DYNAMIC QUANTUM USING THE MEAN AVERAGE. IJCSI INTERNATIONAL JOURNAL OF COMPUTER SCIENCE ISSUES.2011 MAY;8(3): 224-29.

[6]. VARMA P S. A BEST POSSIBLE TIME QUANTUMFOR IMPROVING SHORTEST REMAINING BURST ROUND ROBIN (SRBRR) ALGORITHM. INTERNATIONAL JOURNAL OF ADVANCED RESEARCH INCOMPUTER SCIENCE AND SOFTWARE.