**EXPERIMENT 6-b**

#include<stdio.h>

int main()

{ int n,at[10], bt[10], ct[10], wt[10], tn =0, c, ta[10],tat[10];

//at-ArritvalTime::br-BurstTime::ct-CompletionTime::ta-TemporaryArray

//wt-WaitingTime::tat-TurnAroundTime::tn-CurrentTime(TimeNow)

// int i, j, k, tot, pc=0, pointer = 0, lp=-1;//lp-Last-executedProcess

int i, k, tot, pc=0, pointer = 0, lp=-1;//lp-Last-executedProcess

char pn[10][10];

printf("Enter the number of processes: ");

scanf("%d",&n);

printf("Enter <ProcessName> <ArrivalTime> <BurstTime>\n");

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

scanf("%s%d%d",&pn[i],&at[i],&bt[i]);

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

{ ct[i] = -1;

ta[i] = bt[i]; }

while(pc!=n)

{ k=0;

c = 0;

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

{ if(ct[i]<0 && at[i]<=tn)

c++; }

if(c==0)

tn++;

else

{ pointer = 0;

while(at[pointer]>tn || ct[pointer]>0)

pointer++;

for(k=pointer+1; k<n; k++)

if( (at[k]<=tn && ct[k]<0) &&

( (bt[pointer]==bt[k] && k==lp) || bt[pointer]>bt[k] ) )

pointer = k;

if(ct[pointer]<0)

{ bt[pointer]--;

tn++;

if(bt[pointer]==0)

{ ct[pointer] = tn;

wt[pointer] = ct[pointer] - ( at[pointer]+ ta[pointer] );

tat[pointer] = ct[pointer] - at[pointer];

pc++; } }

lp = pointer; } }

printf("\nPN\tAT\tBT\tCT\tWT\tTAT\n");

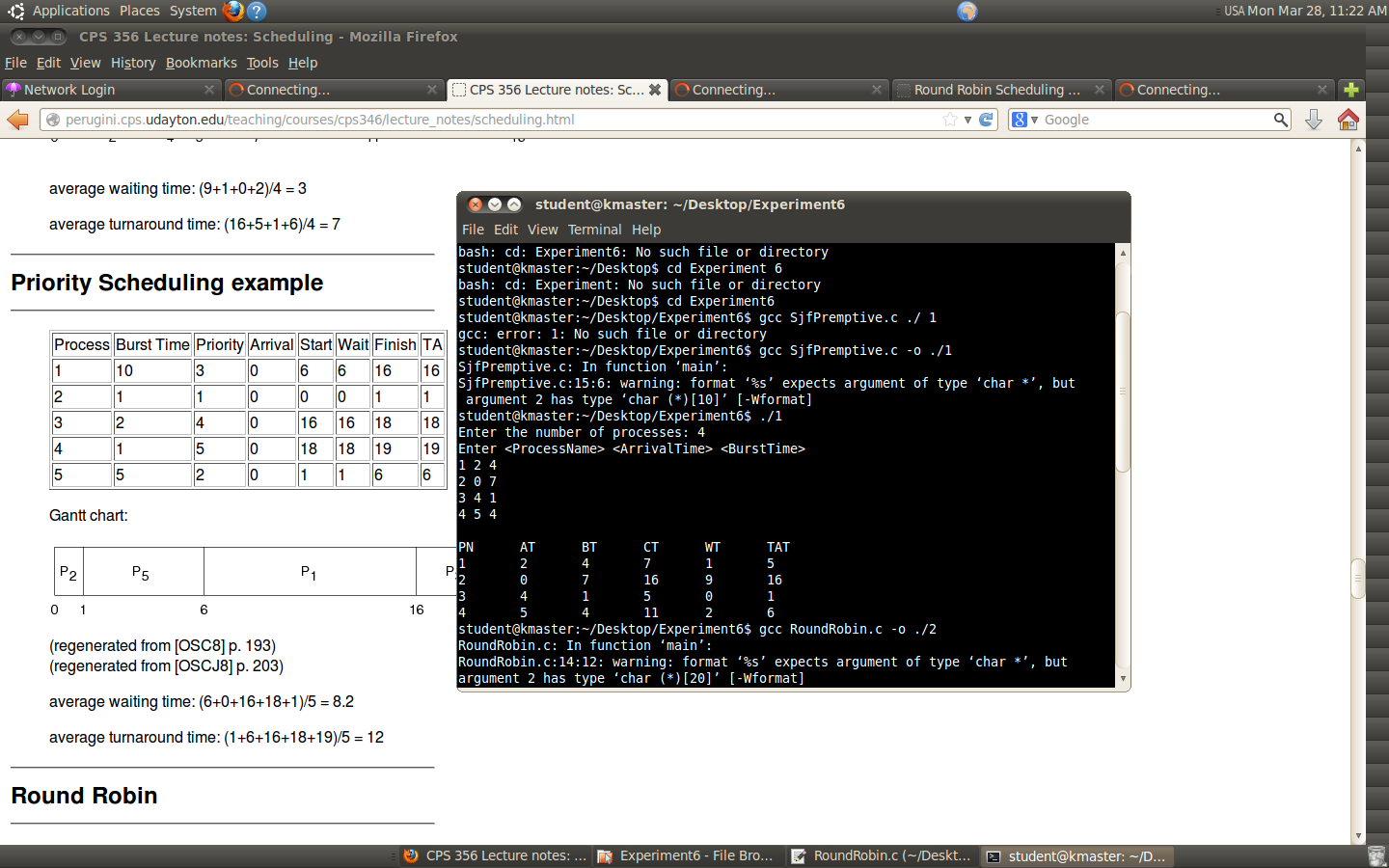
for(i=0;i<n;i++)

printf("%s\t%d\t%d\t%d\t%d\t%d\n",pn[i],at[i],ta[i],ct[i],wt[i],tat[i]);

return 0;

}

**OUTPUT:**

****

**EXPERIMENT 6-d**

PRIORITY SCHEDULING

#include<stdio.h>

#include<string.h>

int main(void)

{ char pn[20][20], c[20][20]; //PN-PROGRAM NAMES C-A TEMPORARY ARRAY

int n,i,j,at[20], bt[20], pt[20], wt[20],ct[20],tat[20];

//bt-BURST TIME ;pt-PRIORITY;wt-WAITING TIME; tat-TURN AROUND TIME

int temp1, temp2, temp3, count=0,twt=0;//,tat=0;

printf("Enter number of processes:");

scanf("%d", &n);

printf("Enter <ProcessName> <ArrivalTime> <BurstTime> <Priority> :\n");

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

scanf("%s%d%d%d",&pn[i],&at[i],&bt[i],&pt[i]);

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

{

for(j=i+1; j<n; j++)

if ( ( (i==0||count<1)&&(at[i]>at[j]||(at[i]==at[j]&&pt[i]>pt[j])) ) || (count == 1 && ct[i-1]>=at[j]) || ((ct[i-1]>=at[j]&&pt[i]>pt[j]))// || (ct[i-1]<at[i]&&ct[i-1]>=at[j])) ) {

temp1 = bt[i];

bt[i] = bt[j];

bt[j] = temp1;

temp2 = at[i];

at[i] = at[j];

at[j] = temp2;

temp3 = pt[i];

pt[i] = pt[j];

pt[j] = temp3;

strcpy(c[i],pn[i]);

strcpy(pn[i],pn[j]);

strcpy(pn[j],c[i]); }

if(i==0 || count<1)

ct[i] = at[i] + bt[i];

else

ct[i] = ct[i-1] + bt[i];

wt[i] = ct[i] - (at[i] + bt[i]);

tat[i] = ct[i] - at[i];

count = 0 ;

for(j=i+1; j<n; j++)

if(ct[i]>=at[j])

count++; }

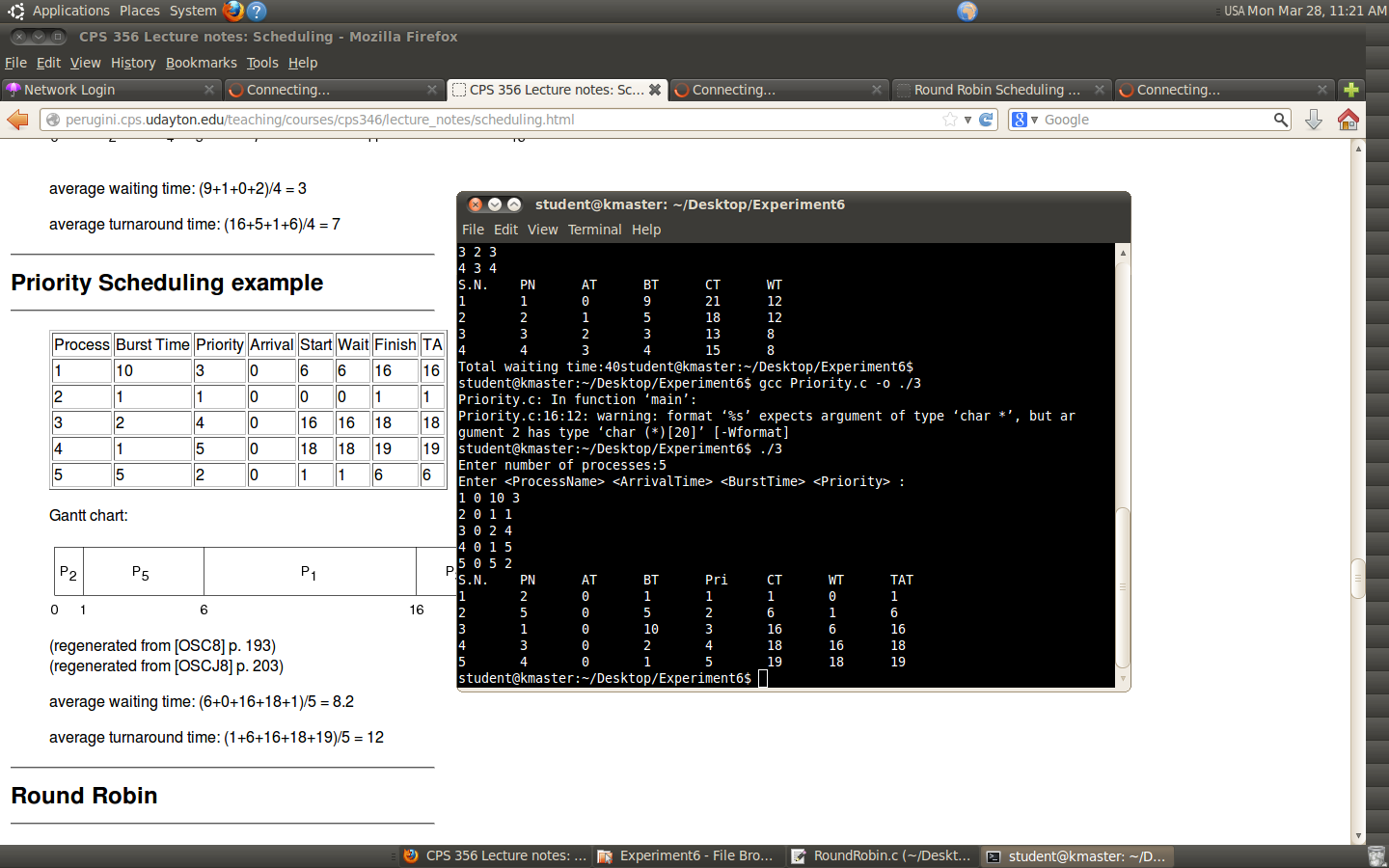
printf("S.N.\tPN\tAT\tBT\tPri\tCT\tWT\tTAT\n");

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

printf("%d\t%s\t%d\t%d\t%d\t%d\t%d\t%d\n",(i+1),pn[i],at[i],bt[i],pt[i],ct[i],wt[i],tat[i]);

}

**OUTPUT:**

****

**EXPERIMENT 6-c**

ROUND ROBIN

#include<stdio.h>

#include<string.h>

int main(void)

{

//VARIABLE DECLARATION

char pn[20][20], c[20][20]; //PN-PROGRAM NAMES

int n,i,j,k,l, tq, at[20], bt[20], rbt[20], wt[20],tt[20],ct[20]; //bt-BURST TIME ; wt-WAITING TIME; tat-TURN AROUND TIME

int temp1, temp2, temp3, count=0,twt=0, tn;//,tat=0;

printf("Enter <Number\_of\_Processes & Time\_Quantum:\n");

scanf("%d%d", &n, &tq);

printf("Enter PN, AT, BT:\n");

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

scanf("%s%d%d",&pn[i],&at[i],&bt[i]);

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

rbt[i]=bt[i];

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

{ for(j=i+1; j<n;j++)

{ if(at[i]>at[j])

{ temp1 = bt[i];

bt[i] = bt[j];

bt[j] = temp1;

temp2 = at[i];

at[i] = at[j];

at[j] = temp2;

temp3 = rbt[i];

rbt[i] = rbt[j];

rbt[j] = temp3;

strcpy(c[i],pn[i]);

strcpy(pn[i],pn[j]);

strcpy(pn[j],c[i]);

}

} //END OF J FOR-LOOP

}//END OF I FOR-LOOP

tn = at[0];

label:

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

{

if(at[i]>tn) i--;

if(rbt[i]>0)

{

if(rbt[i]>tq)

{

tn += tq;

rbt[i] -= tq;

}

else

{

tn += rbt[i];

rbt[i] = 0;

ct[i] = tn;

count++;

}

}

}

if(count<n) goto label;

//CALCULATING WAITING TIME & TAT

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

{

wt[i] = ct[i]-at[i]-bt[i];

twt += wt[i];

}

//PRINTING THE VALUES AFTER ALL PREOCESSES COMPLETED

printf("S.N.\tPN\tAT\tBT\tCT\tWT\n");

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

printf("%d\t%s\t%d\t%d\t%d\t%d\n",(i+1),pn[i],at[i],bt[i],ct[i],wt[i]);

printf("Total waiting time:%d", twt);

}

**OUTPUT:**

