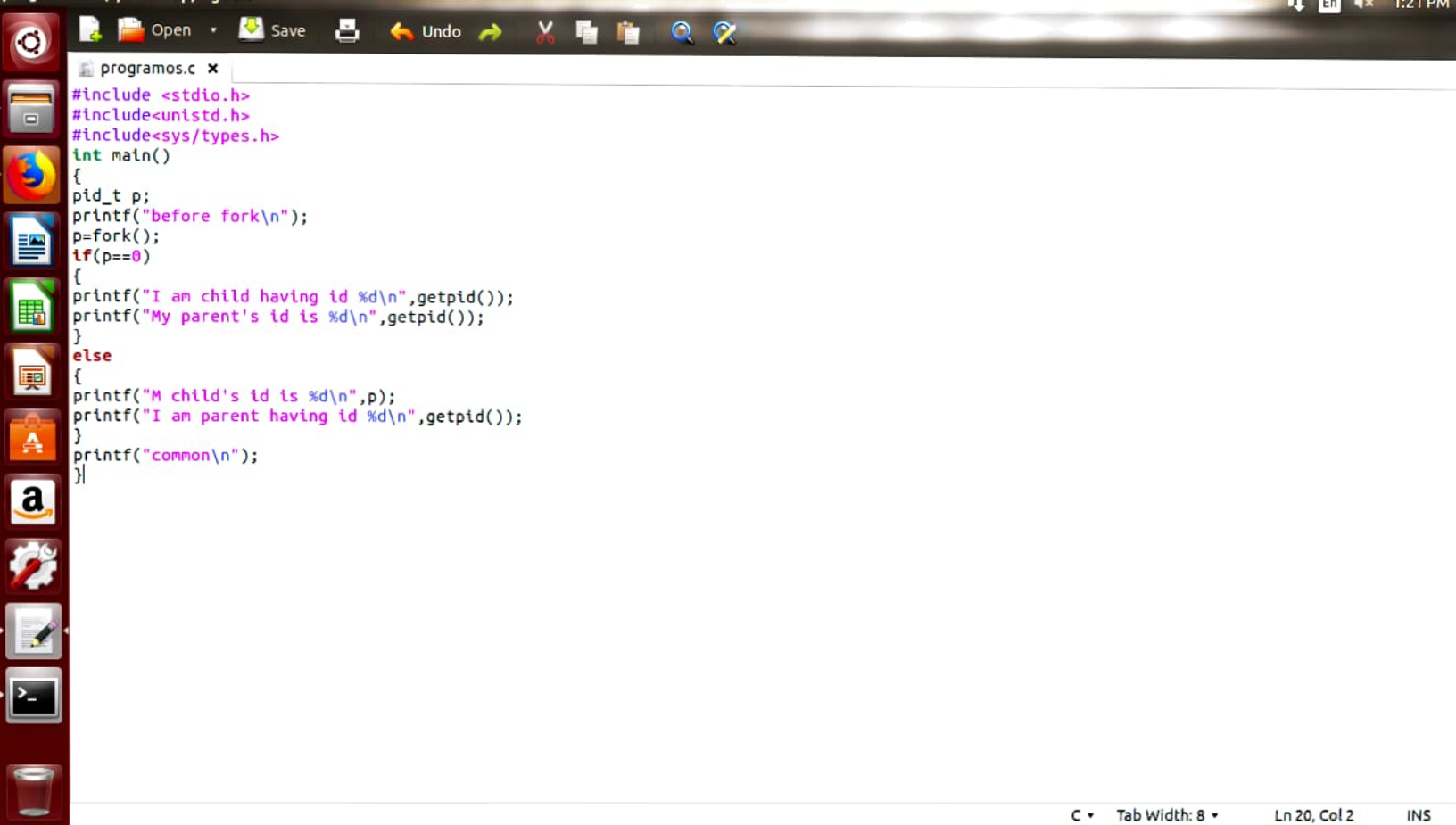
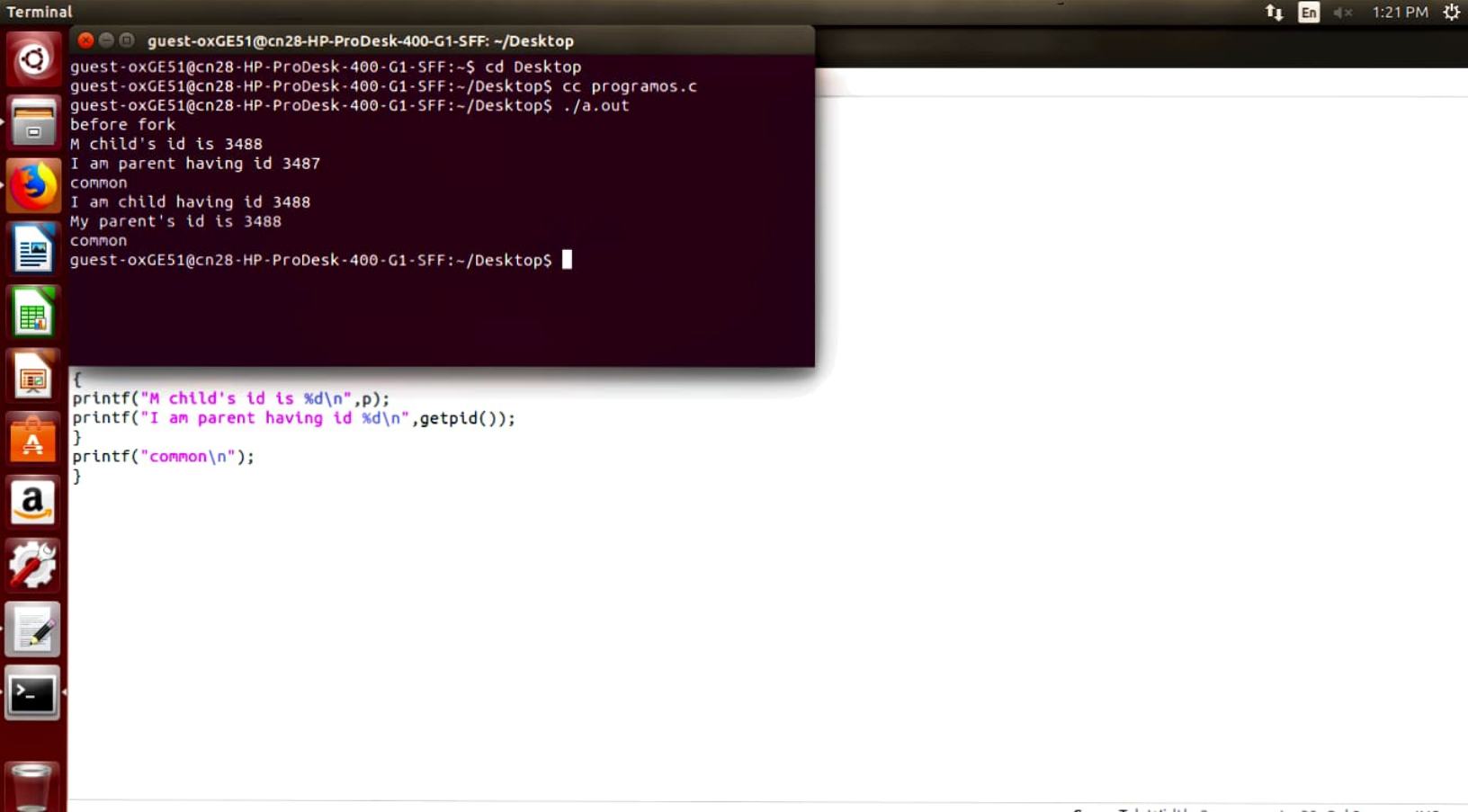
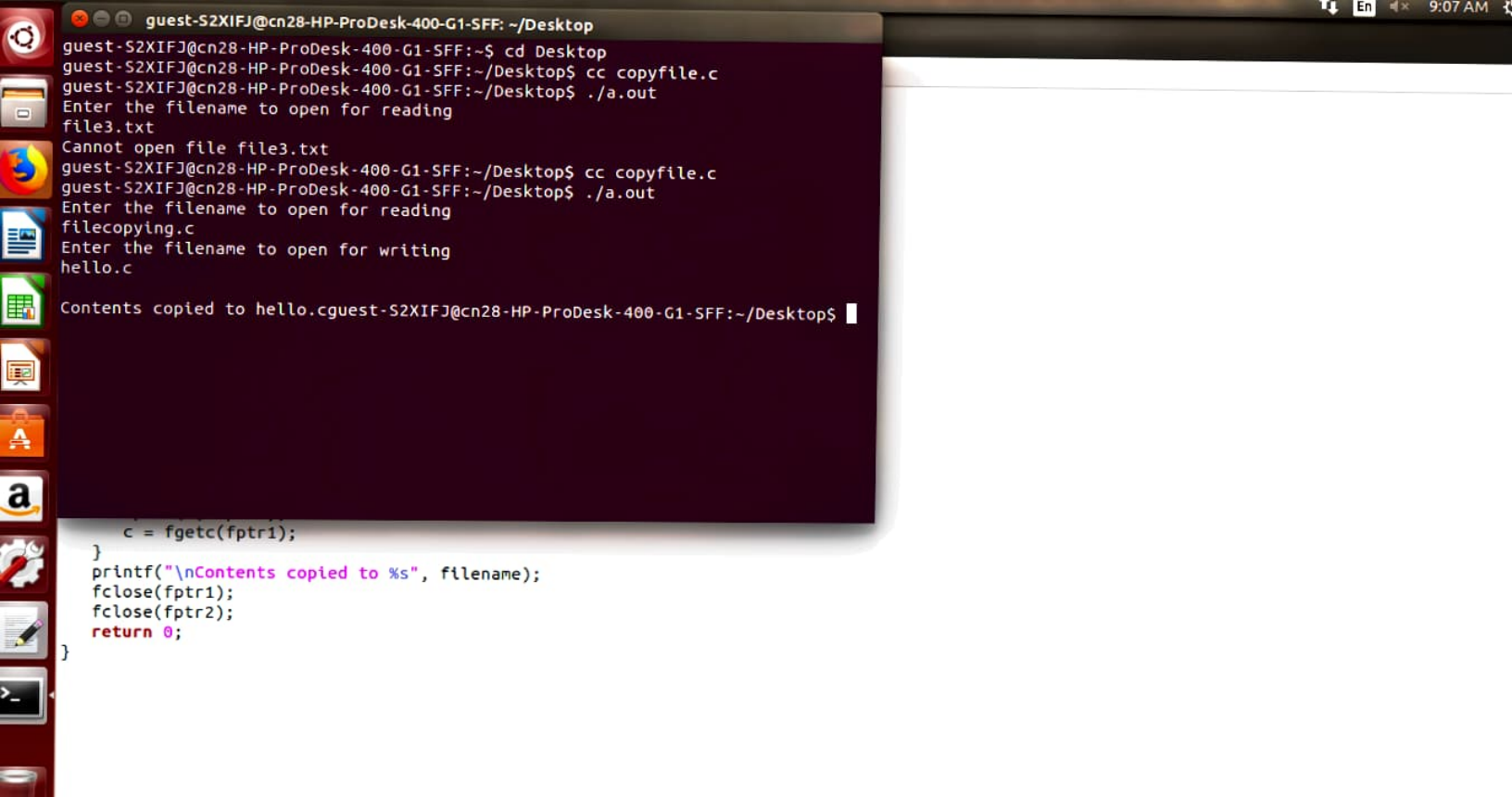

Open
Save
Undo
programos.c
#include <stdio.h>
#include<unistd.h>
#include<sys/types.h>
int main()
{
pid_t p;
printf("before fork\n");
p=fork();
if(p==0)
{
printf("I am child having id %d\n",getpid());
printf("My parent's id is %d\n",getpid());
}
else
{
printf("M child's id is %d\n",p);
printf("I am parent having id %d\n",getpid());
}
printf("common\n");
}
C
Tab Width: 8
Ln 20, Col 2
INS


Terminal
1:21 PM
guest-oxGE51@cn28-HP-ProDesk-400-G1-SFF: ~/Desktop
guest-oxGE51@cn28-HP-ProDesk-400-G1-SFF:~$ cd Desktop
guest-oxGE51@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$ cc programos.c
guest-oxGE51@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$ ./a.out
before fork
M child's id is 3488
I am parent having id 3487
common
I am child having id 3488
My parent's id is 3488
common
guest-oxGE51@cn28-HP-ProDesk-400-G1-SFF:~/Desktop$
{
printf("M child's id is %d\n",p);
printf("I am parent having id %d\n",getpid());
}
printf("common\n");
}

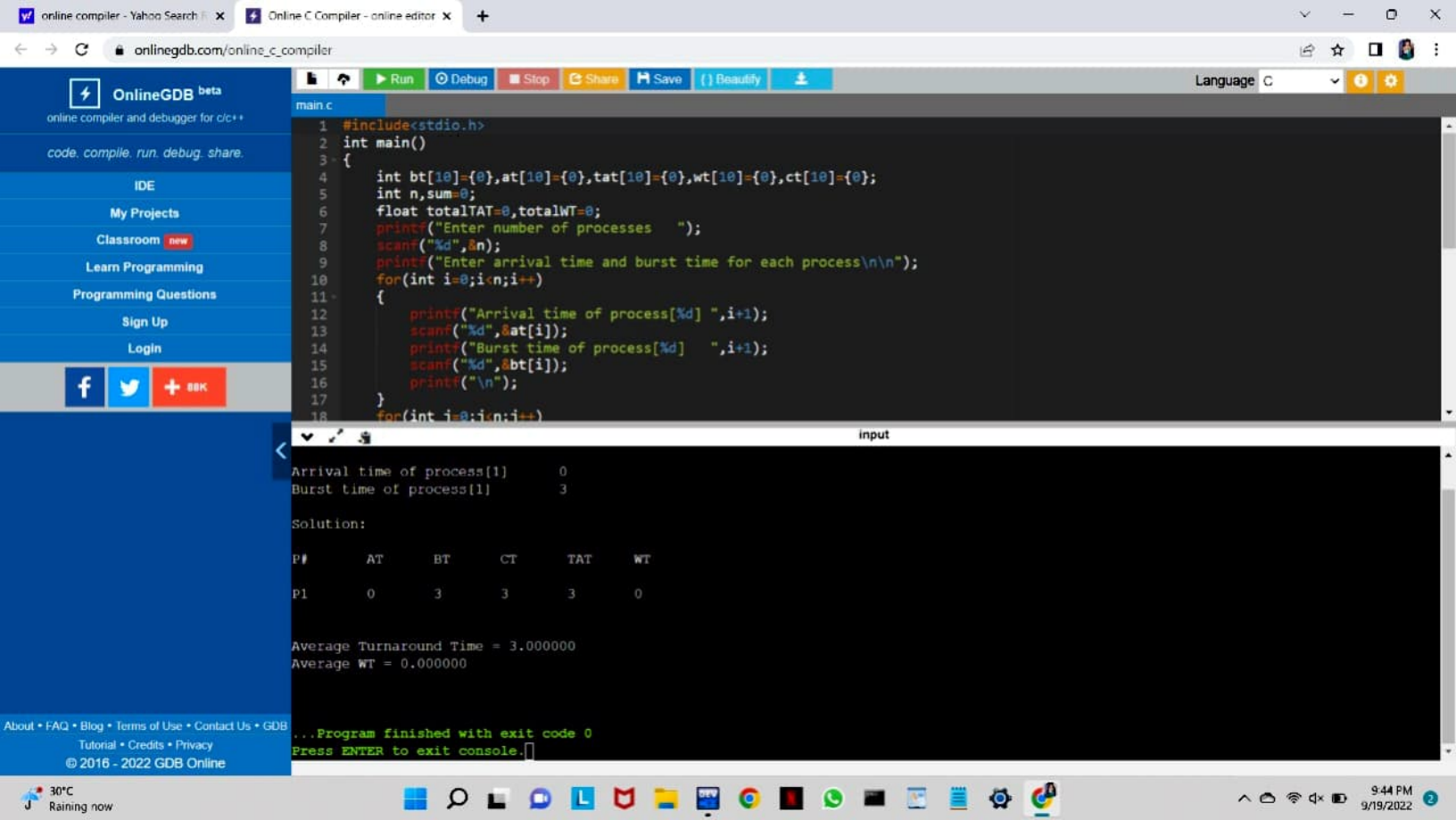
Open Save Undo

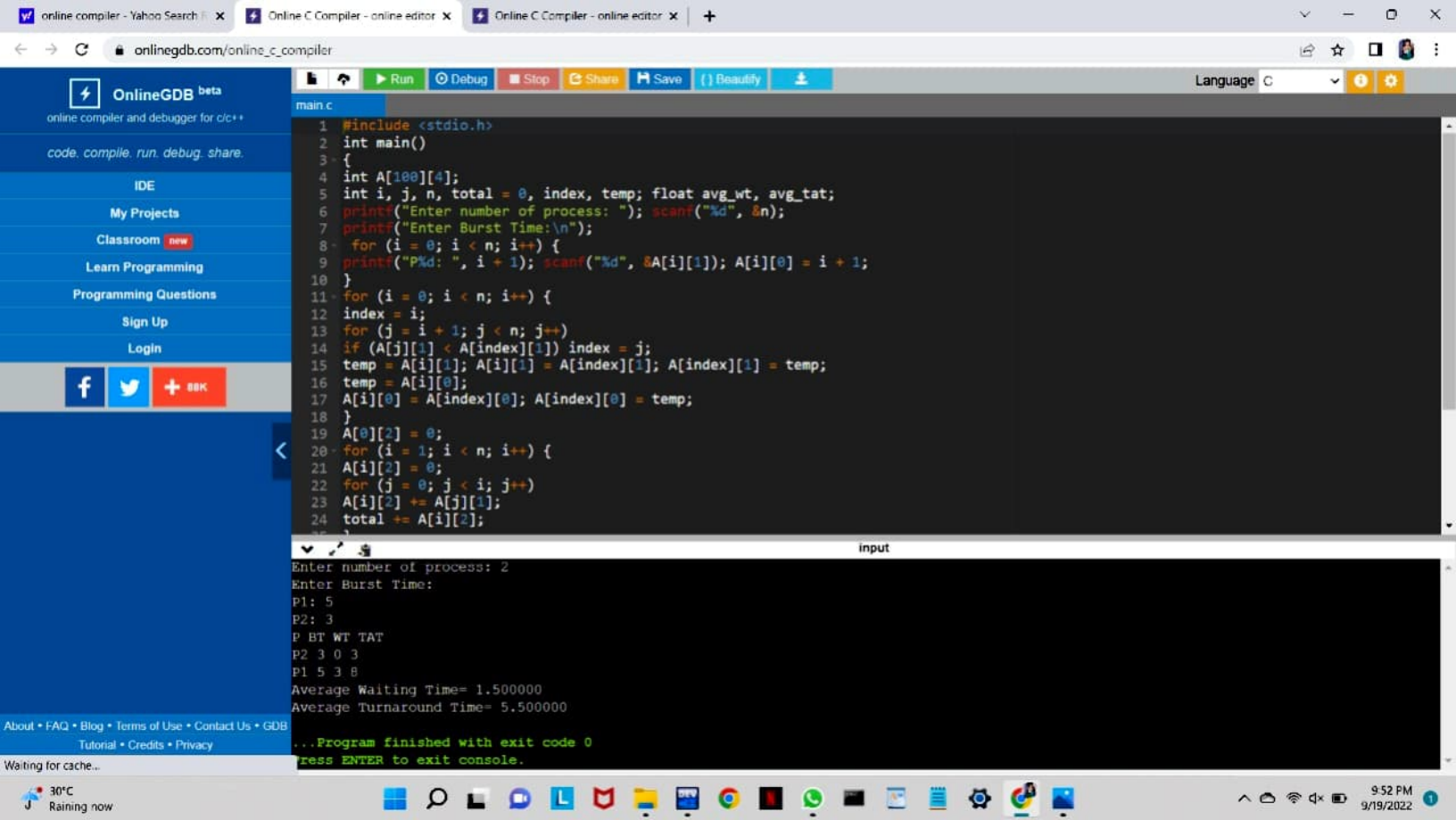
filecopying.c × copyfile.c × hello.c ×

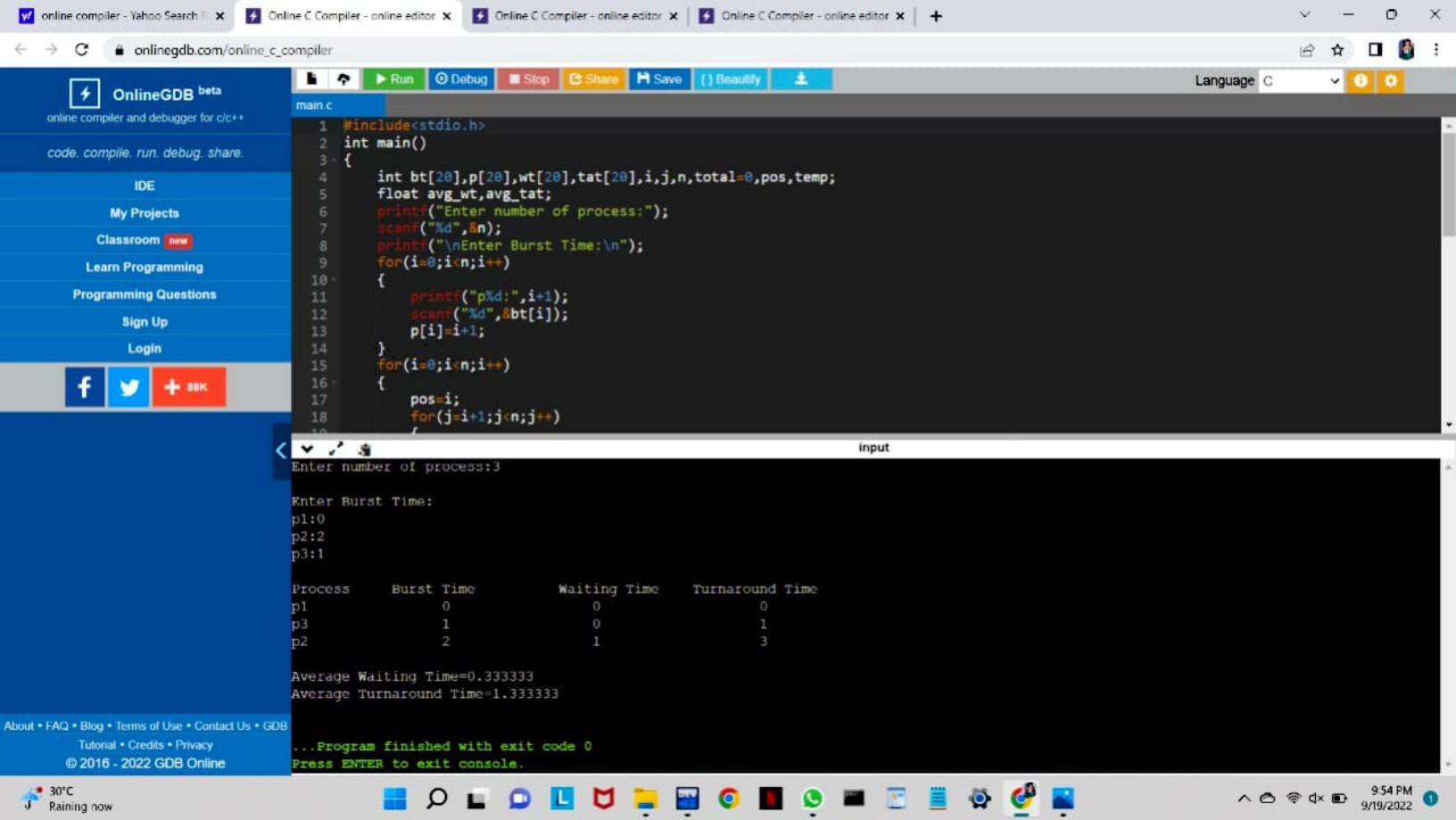
```c
#include <stdio.h>
#include <stdlib.h>
int main(){
   FILE *fptr1, *fptr2;
   char filename[100], c;
   printf("Enter the filename to open for reading \n");
   scanf("%s",filename);
   fptr1 = fopen(filename, "r");
   if (fptr1 == NULL){
      printf("Cannot open file %s \n", filename);
      exit(0);
   }
   printf("Enter the filename to open for writing \n");
   scanf("%s", filename);
   fptr2 = fopen(filename, "w");
   if (fptr2 == NULL){
      printf("Cannot open file %s \n", filename);
      exit(0);
   }
   c = fgetc(fptr1);
   while (c != EOF){
      fputc(c, fptr2);
      c = fgetc(fptr1);
   }
   printf("\nContents copied to %s", filename);
   fclose(fptr1);
   fclose(fptr2);
   return 0;
}
```




```
    c = fgetc(fptr1);
  }
  printf("\nContents copied to %s", filename);
  fclose(fptr1);
  fclose(fptr2);
  return 0;
}
```


online compiler - Yahoo Search
Online C Compiler - online editor
onlinegdb.com/online_c_compiler
OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
88K
About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy
© 2016 - 2022 GDB Online
Run
Debug
Stop
Share
Save
{ } Beautify
Language C
main.c
#include<stdio.h>
int main()
{
int bt[10]={0},at[10]={0},tat[10]={0},wt[10]={0},ct[10]={0};
int n,sum=0;
float totalTAT=0,totalWT=0;
printf("Enter number of processes ");
scanf("%d",&n);
printf("Enter arrival time and burst time for each process\n\n");
for(int i=0;i<n;i++)
{
printf("Arrival time of process[%d] ",i+1);
scanf("%d",&at[i]);
printf("Burst time of process[%d] ",i+1);
scanf("%d",&bt[i]);
printf("\n");
}
for(int i=0;i<n;i++)
input
Arrival time of process[1] 0
Burst time of process[1] 3
Solution:
P# AT BT CT TAT WT
P1 0 3 3 3 0
Average Turnaround Time = 3.000000
Average WT = 0.000000
...Program finished with exit code 0
Press ENTER to exit console.
30°C
Raining now
9:44 PM
9/19/2022


online compiler - Yahoo Search R
Online C Compiler - online editor
Online C Compiler - online editor
onlinegdb.com/online_c_compiler
OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
88K
Run
Debug
Stop
Share
Save
{} Beautify
Language C
main.c
#include <stdio.h>
int main()
{
int A[100][4];
int i, j, n, total = 0, index, temp; float avg_wt, avg_tat;
printf("Enter number of process: "); scanf("%d", &n);
printf("Enter Burst Time:\n");
for (i = 0; i < n; i++) {
printf("P%d: ", i + 1); scanf("%d", &A[i][1]); A[i][0] = i + 1;
}
for (i = 0; i < n; i++) {
index = i;
for (j = i + 1; j < n; j++)
if (A[j][1] < A[index][1]) index = j;
temp = A[i][1]; A[i][1] = A[index][1]; A[index][1] = temp;
temp = A[i][0];
A[i][0] = A[index][0]; A[index][0] = temp;
}
A[0][2] = 0;
for (i = 1; i < n; i++) {
A[i][2] = 0;
for (j = 0; j < i; j++)
A[i][2] += A[j][1];
total += A[i][2];
input
Enter number of process: 2
Enter Burst Time:
P1: 5
P2: 3
P BT WT TAT
P2 3 0 3
P1 5 3 8
Average Waiting Time= 1.500000
Average Turnaround Time= 5.500000
...Program finished with exit code 0
ress ENTER to exit console.
About • FAQ • Blog • Terms of Use • Contact Us • GDB
Tutorial • Credits • Privacy
Waiting for cache...
30°C
Raining now
9:52 PM
9/19/2022


online compiler - Yahoo Search
Online C Compiler - online editor
Online C Compiler - online editor
Online C Compiler - online editor
onlinegdb.com/online_c_compiler
OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
88K
Run
Debug
Stop
Share
Save
{} Beautify
Language C
main.c
#include<stdio.h>
int main()
{
int bt[20],p[20],wt[20],tat[20],i,j,n,total=0,pos,temp;
float avg_wt,avg_tat;
printf("Enter number of process:");
scanf("%d",&n);
printf("\nEnter Burst Time:\n");
for(i=0;i<n;i++)
{
printf("p%d:",i+1);
scanf("%d",&bt[i]);
p[i]=i+1;
}
for(i=0;i<n;i++)
{
pos=i;
for(j=i+1;j<n;j++)
input
Enter number of process:3
Enter Burst Time:
p1:0
p2:2
p3:1
Process Burst Time Waiting Time Turnaround Time
p1 0 0 0
p3 1 0 1
p2 2 1 3
Average Waiting Time=0.333333
Average Turnaround Time=1.333333
...Program finished with exit code 0
Press ENTER to exit console.
About • FAQ • Blog • Terms of Use • Contact Us • GDB
Tutorial • Credits • Privacy
© 2016 - 2022 GDB Online
30°C
Raining now
9:54 PM
9/19/2022