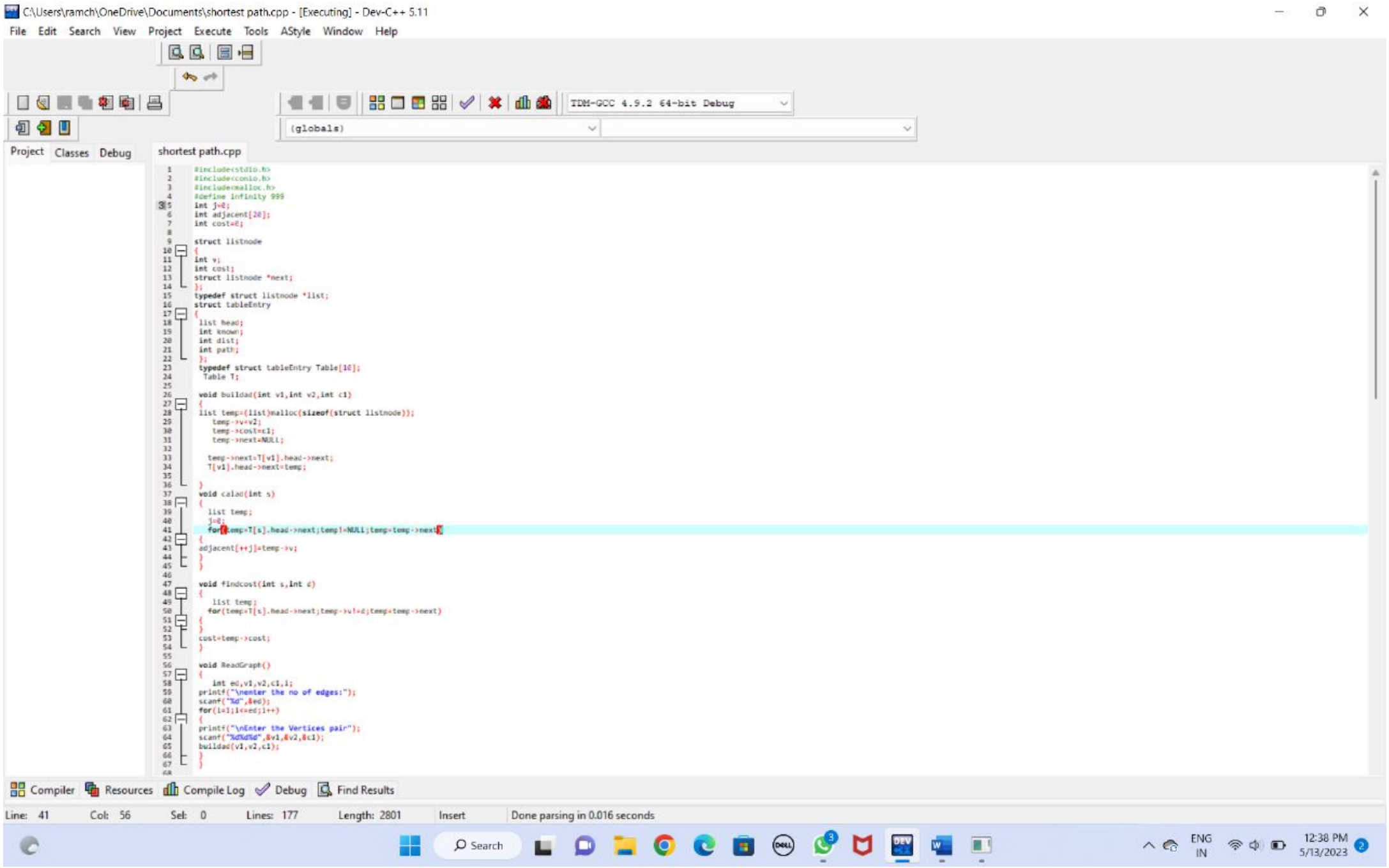
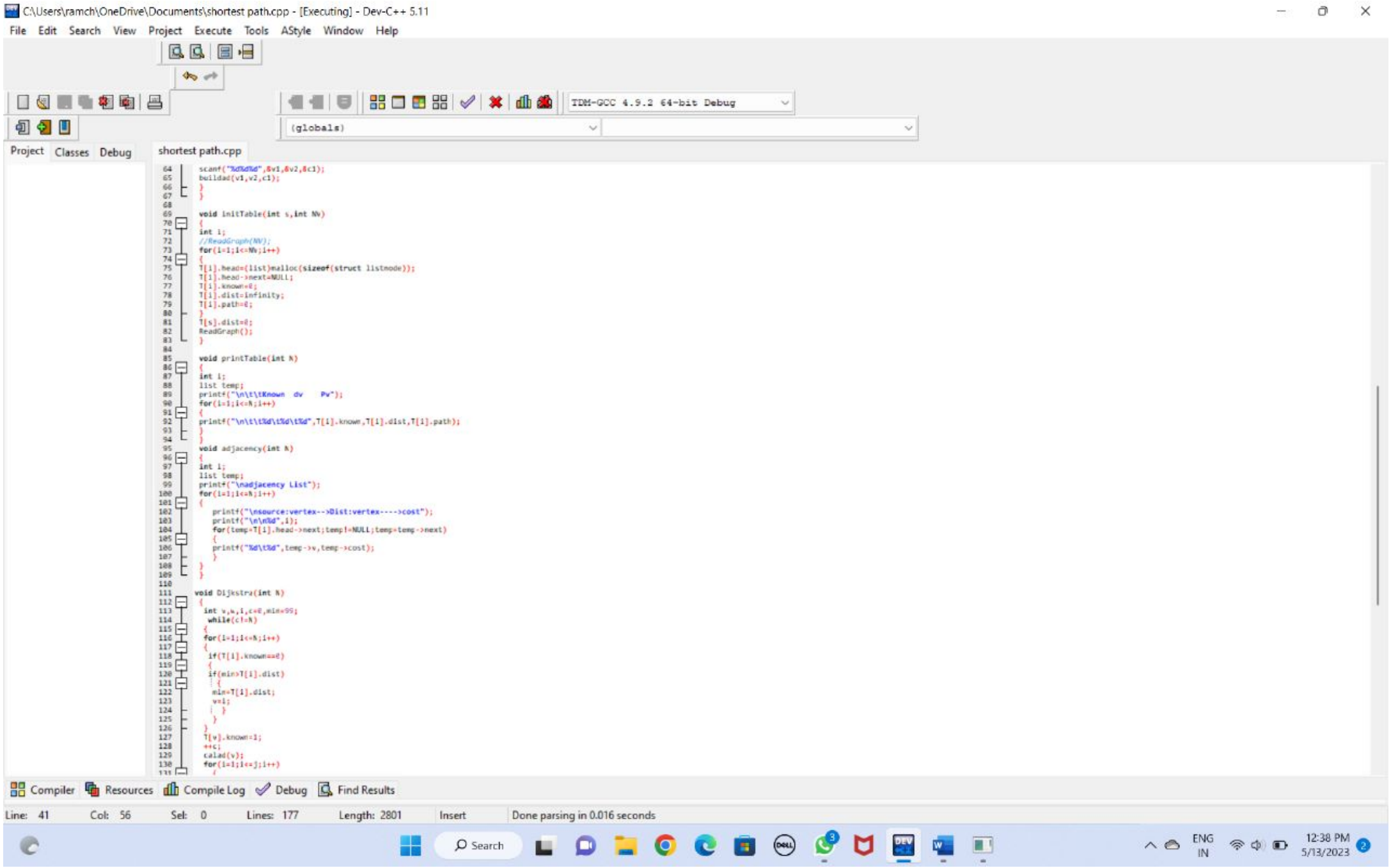
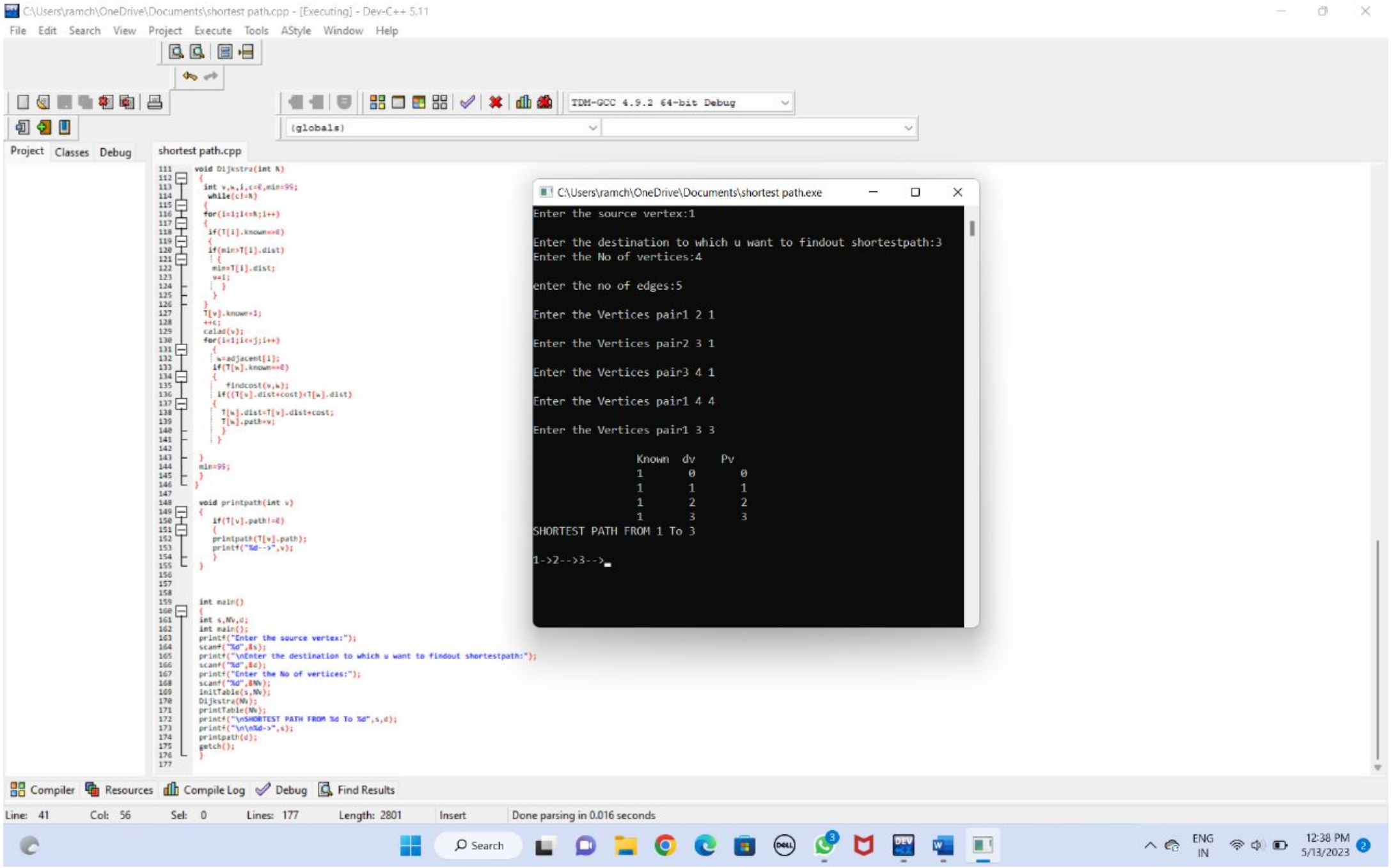

C:\Users\ramch\OneDrive\Documents\shortest path.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Debug
(globals)
Project Classes Debug
shortest path.cpp
#include<stdio.h>
#include<conio.h>
#include<malloc.h>
#define Infinity 999
int j=0;
int adjacent[20];
int cost=0;
struct listnode
{
int v;
int cost;
struct listnode *next;
};
typedef struct listnode *list;
struct tableEntry
{
list head;
int known;
int dist;
int path;
};
typedef struct tableEntry Table[10];
Table T;
void buildad(int v1,int v2,int c1)
{
list temp=(list)malloc(sizeof(struct listnode));
temp->v=v2;
temp->cost=c1;
temp->next=NULL;
temp->next=T[v1].head->next;
T[v1].head->next=temp;
}
void calad(int s)
{
list temp;
j=0;
for(temp=T[s].head->next;temp!=NULL;temp=temp->next)
{
adjacent[++j]=temp->v;
}
}
void findcost(int s,int d)
{
list temp;
for(temp=T[s].head->next;temp->v!=d;temp=temp->next)
{
}
cost=temp->cost;
}
void ReadGraph()
{
int ed,v1,v2,c1,i;
printf("\nenter the no of edges:");
scanf("%d",&ed);
for(i=1;i<=ed;i++)
{
printf("\nEnter the Vertices pair");
scanf("%d%d%d",&v1,&v2,&c1);
buildad(v1,v2,c1);
}
}
Compiler Resources Compile Log Debug Find Results
Line: 41 Col: 56 Sel: 0 Lines: 177 Length: 2801 Insert Done parsing in 0.016 seconds
Search
ENG IN
12:38 PM 5/13/2023


C:\Users\ramch\OneDrive\Documents\shortest path.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Debug
(globals)
Project Classes Debug
shortest path.cpp
scanf("%d%d%d",&v1,&v2,&c1);
buildad(v1,v2,c1);
}
}
void initTable(int s,int Nv)
{
int i;
//ReadGraph(NV);
for(i=1;i<=Nv;i++)
{
T[i].head=(list)malloc(sizeof(struct listnode));
T[i].head->next=NULL;
T[i].known=0;
T[i].dist=infinity;
T[i].path=0;
}
T[s].dist=0;
ReadGraph();
}
void printTable(int N)
{
int i;
list temp;
printf("\n\t\tKnown dv Pv");
for(i=1;i<=N;i++)
{
printf("\n\t\t%d\t%d\t%d",T[i].known,T[i].dist,T[i].path);
}
}
void adjacency(int N)
{
int i;
list temp;
printf("\nadjacency List");
for(i=1;i<=N;i++)
{
printf("\nsource:vertex-->Dist:vertex---->cost");
printf("\n\n%d",i);
for(temp=T[i].head->next;temp!=NULL;temp=temp->next)
{
printf("%d\t%d",temp->v,temp->cost);
}
}
}
void Dijkstra(int N)
{
int v,w,i,c=0,min=99;
while(c!=N)
{
for(i=1;i<=N;i++)
{
if(T[i].known==0)
{
if(min>T[i].dist)
{
min=T[i].dist;
v=i;
}
}
}
T[v].known=1;
++c;
calad(v);
for(i=1;i<=j;i++)
Compiler Resources Compile Log Debug Find Results
Line: 41 Col: 56 Sel: 0 Lines: 177 Length: 2801 Insert Done parsing in 0.016 seconds
Search
ENG IN
12:38 PM 5/13/2023


C:\Users\ramch\OneDrive\Documents\shortest path.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Debug
(globals)
Project Classes Debug
shortest path.cpp
void Dijkstra(int N)
{
int v,w,i,c=0,min=99;
while(c!=N)
{
for(i=1;i<=N;i++)
{
if(T[i].known==0)
{
if(min>T[i].dist)
{
min=T[i].dist;
v=i;
}
}
}
T[v].known=1;
++c;
calad(v);
for(i=1;i<=j;i++)
{
w=adjacent[i];
if(T[w].known==0)
{
findcost(v,w);
if((T[v].dist+cost)<T[w].dist)
{
T[w].dist=T[v].dist+cost;
T[w].path=v;
}
}
}
min=99;
}
}
void printpath(int v)
{
if(T[v].path!=0)
{
printpath(T[v].path);
printf("%d-->",v);
}
}
int main()
{
int s,Nv,d;
int main();
printf("Enter the source vertex:");
scanf("%d",&s);
printf("\nEnter the destination to which u want to findout shortestpath:");
scanf("%d",&d);
printf("Enter the No of vertices:");
scanf("%d",&Nv);
InitTable(s,Nv);
Dijkstra(Nv);
printTable(Nv);
printf("\nSHORTEST PATH FROM %d To %d",s,d);
printf("\n\n%d->",s);
printpath(d);
getch();
}
C:\Users\ramch\OneDrive\Documents\shortest path.exe
Enter the source vertex:1
Enter the destination to which u want to findout shortestpath:3
Enter the No of vertices:4
enter the no of edges:5
Enter the Vertices pair1 2 1
Enter the Vertices pair2 3 1
Enter the Vertices pair3 4 1
Enter the Vertices pair1 4 4
Enter the Vertices pair1 3 3
Known dv Pv
1 0 0
1 1 1
1 2 2
1 3 3
SHORTEST PATH FROM 1 To 3
1->2-->3-->
Compiler Resources Compile Log Debug Find Results
Line: 41 Col: 56 Sel: 0 Lines: 177 Length: 2801 Insert Done parsing in 0.016 seconds
Search
ENG IN
12:38 PM 5/13/2023