Name : Patil Priyanka Nimba

Email Id: [patilpriyanka223@gamil.com](mailto:patilpriyanka223@gamil.com)

Mobile no:8788308292

Question 2 :**Calculate the surface of the building exposed to sunlight**

**# Program1:Simple program (only one building and positive value)**

**#include** <iostream>

**using** **namespace** std;

**struct** Point

{

**double** x;

**double** y;

};

**float** **countLen**(Point building[],Point p)

{

**float** l,w,h;

**if**((p.x<building[0].x)&&(p.y<building[0].y))

{

w=building[3].x-building[0].x;

h=building[1].y-building[0].y;

l=w+h;

}**else** **if**(p.x<building[0].x && p.y>building[0].y && p.y<building[1].y )

{

h=building[1].y-building[0].y;

l=h;

}**else** **if**(p.x<building[1].x &&p.y>building[1].y)

{

w=building[2].x-building[0].x;

h=building[1].y-building[0].y;

l=w+h;

}**else** **if**(p.x>building[0].x && p.y>building[1].y && p.y<building[2].y)

{

w=building[2].x-building[1].x;

l=w;

}**else** **if**(p.x>building[3].x &&p.y>building[2].y)

{

w=building[2].x-building[1].x;

h=building[2].y-building[3].y;

l=w+h;

}**else** **if**(p.x>building[3].x && p.y>building[3].y && p.y<building[2].y)

{

h=building[3].y-building[3].y;

l=h;

}**else** **if**(p.x>building[3].x &&p.y<building[2].y)

{

w=building[2].x-building[1].x;

h=building[2].y-building[3].y;

l=w+h;

}**else** **if**(p.x>building[0].x && p.x<building[3].x && p.y<building[3].y)

{

w=building[3].x-building[01].x;

l=w;

}

**return** l;

}

**int** **main**()

{

**struct** Point p;

**struct** Point building1[4];

**for**(**int** j=0;j<4;j++)

{

cout<<"Enter X coordinate of "<<j+1<<"point of building"<<**endl**;

cin>>building1[j].x;

cout<<"Enter Y coordinate of "<<j+1<<"point of building"<<**endl**;

cin>>building1[j].y;

}

cout<<"Enter the source point"<<**endl**;

cout<<"Enter the X coordinate of source"<<**endl**;

cin>>p.x;

cout<<"Enter the Y coordinate of source"<<**endl**;

cin>>p.y;

**float** length=countLen(building1, p);

cout<<"Length"<<length<<**endl**;

**return** 0;

}

---------------------------------------------------------------------------------------------------------------------# program 2:Multiple buildings and values

**#include** <iostream>

**using** **namespace** std;

**struct** Point

{

**double** x;

**double** y;

};

**float** **position**(Point (\*building)[4],**float** w,**float** h,Point p) //position of source p

{

**float** len,w1,h1;

**int** n=**sizeof**(building)/**sizeof**(building[0]); //number of buildings

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

{

**for**(**int** j=0;j<4;j++)

{

**if**(p.x <building[i][j].x && p.y<building[i][j].y)

{

w1=w[i];

h1=h[i];

len=w1+h1;

}**else** **if**( p.x <building[i][j].x && p.y<building[i][j].y&&p.y>building[i][1].y )

{

h1=h[i];

len=h1;

}**else** **if**(p.x <building[i][j].x && p.y>building[i][j].y)

{

w1=w[i];

h1=h[i];

len=w1+h1;

}**else** **if**( p.x >building[i][1].x && p.y>building[i][1].y&&p.y>building[i][2].y )

{

len=w[i];

}**else** **if**(p.x >building[i][2].x && p.y>building[i][3].y)

{

len=w[i]+h[i];

}**else** **if**( p.x >building[i][4].x && p.y>building[i][3].y&&p.y<building[i][2].y )

{

len=h[1];

}

**else** **if**(p.x >building[i][3].x && p.y<building[i][3].y)

{

len=w[i]+h[i];

}**else** **if**( p.x >building[i][0].x && p.x>building[i][3].x&&p.y<building[i][0].y )

{

len=w[1];

}

}

}

**return** len;

}

**float** **countLen**(Point (\*building)[4],Point p)

{

**int** n=**sizeof**(building)/**sizeof**(building[0]); //calculate number of building.

**float** w[n],h[n],temp,len; //store height and weight of building.

w[0]=0.0;

h[0]=0.0;

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

{

**for**(**int** j=0;j<4;j++)

{

temp=0.0;

temp=building[i][j].x-building[i][j+1].x;

w[i]= max(w[i],temp); //maximum weigh value of building

temp=0.0;

temp=building[i][j].y-building[i][j+1].y;

h[i]=max(h[i],temp); //maximum height value of building

}

}

len=position(building,w,h,p);

**return** len;

}

**int** **main**() {

**int** n;

**struct** Point p;

cout<<"Enter the number of buildings"<<**endl**;

cin>>n;

**struct** Point \*\*building1=**new** **struct** Point \*[n];

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

{

building1[i]=**new** **struct** Point[4];

}

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

{

cout<<"Enter "<<i+1<<"th building coordinates"<<**endl**;

**for**(**int** j=0;j<4;j++)

{

cout<<"Enter X coordinate of "<<j+1<<"point of building"<<**endl**;

cin>>building1[i][j].x;

cout<<"Enter Y coordinate of "<<j+1<<"point of building"<<**endl**;

cin>>building1[i][j].y;

}

}

cout<<"Enter the source point"<<**endl**;

cout<<"Enter the X coordinate of source"<<**endl**;

cin>>p.x;

cout<<"Enter the Y coordinate of source"<<**endl**;

cin>>p.y;

**float** length=countLen(building1,p);

cout<<"Length= "<<length<<**endl**;

**return** 0;

}