University of Burgundy

Software Engineering

Tutorial 4

Lab Report-4

Author:
KODIPAKA VAMSHI

Supervisor:
Dr. Yohan Fougerolle

November 5, 2018

1 Point2D

```
#include <iostream>
using namespace std;
struct Point2D
{
 //   char name[50];
    float x;
    float y;
};
void display(Point2D p1){
//    cout << "Point Name: " << p1.name << endl;
    cout <<"x: " << p1.x << endl;
    cout <<"y: " << p1.y << endl;
}
void display(Point2D);
void display1(Point2D (*ptr)){
//    cout << "Point Name: " << (*ptr).name << endl;
    cout <<"x: " << (*ptr).x << endl;
    cout <<"y: " << (*ptr).y << endl;
}
void display1(Point2D);

 void BuildPoint(Point2D);
void BuildPoint(Point2D (*p2)){
//    cout << "Enter Point name: ";
//    cin.get(p2->name, 50); cout << endl;
    cout << "Enter x cordinate: ";
    cin >> (p2->x);
    cout << "Enter y cordinate: ";
    cin >> (p2->y); cout << endl;
}
```

```
 main()
{
    Point2D *ptr, p1, *ptr2, p2;
   // Point2D dummy={"a",10,9};
    Point2D dummy;
  //  dummy.name[50] = 'dummy';
    dummy.x = 10.3;
    dummy.y = 9.88;
    cout << "Enter Point Information: "<<endl;
    //cin.get(p1.name, 50);
    cout << "Enter x cordinate: ";  cin >> p1.x;
    cout << "Enter y cordinate: ";  cin >> p1.y;
```

```
 //display point2d----------------------
    cout << "\nDisplaying Point Information." << endl;
    display(p1);
    cout << endl;
//display point2d by pointer1----------------------
     ptr = &p1;
     cout << "Displaying point info by pointers." << endl;
    display1(ptr);
  //cout << (*ptr).name << "  " << (*ptr).x << " "<< (*ptr).y << " ";
    cout << endl;

     ptr2 = &p2;
 //Reading Point2D Information by pointer2----------------
   cout << "\nReading Point Info by pointer2.
    (BuildPoint::User i/p function)" << endl;
    BuildPoint(ptr2);
    cout << endl;
 //display point2d by pointer2----------------------
     cout << "Displaying point info by pointer2.
     (BuildPoint::User i/p function)" << endl;
     display1(ptr2);
     //cout << (*ptr).name << "  " << (*ptr).x << " "<< (*ptr).y << " ";
     cout << endl;

       cout << "Dummy structure is created already"<<endl;
//     cout << " dummy Point name: " <<dummy.name<< endl;;
       cout << " dummy x cordinate: "<<dummy.x<<endl;
       cout << " dummy y cordinate: "<<dummy.y<<endl;
}
```

2 Polygon2d

```
#include <iostream>

using namespace std;

struct Polygon2D
{
    float x;
    float y;
    Polygon2D *prev;
    Polygon2D *next;
};

Polygon2D* BuildPolygon( int n)
```

```
{
    Polygon2D *Head = NULL;
    Polygon2D *Newnode=NULL,*temp=NULL;
    for(int i=0; i<n;i++)
    {
        Newnode = new Polygon2D;
        cout << "Enter x cordinate: "; cin>> Newnode->x;
        cout << "Enter y cordinate: "; cin>> Newnode->y;

        if (Head==NULL)
        {

            //intializing the node
            Head=Newnode;
            Newnode->next=Head;
            Newnode->prev=Head;

        }
        else
        {

            //intializing the node for second iteration

            Newnode-> prev = Head -> prev;
            Head -> prev -> next = Newnode;
            Newnode->next=Head;
            Head -> prev = Newnode;
        }
    }
    return Head;

}

void display(Polygon2D *Head, int n)
{
    Polygon2D *p=Head;
    for(int i=0; i<n;i++)
    {
        cout << "Current Points: " << endl;
        cout <<"x: " << p->x ;
        cout <<", y: " << p->y << endl;
        cout << "next Points: " << endl;
        cout <<"x: " << p->next->x ;
        cout <<", y: " << p->next->y << endl;
        cout << "Previous Points: " << endl;
```

```
        cout <<"x: " << p->prev->x ;
        cout <<", y: " << p->prev->y << endl;
        p=p->next;
    }


}
```

```
int main()
{
    int n;
    cout<<"Enter the no of points: ";
    cin>>n;
    Polygon2D *Head;
    Head = BuildPolygon(n);
    display(Head,3);
}
```

3 Polygon2D :: insert,delete a Point2D

```
#include <iostream>

using namespace std;

struct Point2D
{
    float x;
    float y;
    Point2D *prev;
    Point2D *next;

};
getelement(){
    Point2D *ptr, p1;
    cout << "Creation of a new Point: "<<endl;
    cout << "Enter x insert cordinate: "; cin >> p1.x;
    cout << "Enter y insert cordinate: "; cin >> p1.y;
    //display point2d by pointer1----------------------

    return(p1.x,p1.y);
}

Point2D* BuildPolygon( int n)
{
```

```
    Point2D *Head = NULL;
    Point2D *Newnode=NULL,*temp=NULL;
    for(int i=0; i<n;i++)
    {

        Newnode = new Point2D;
        cout << "Enter x cordinate: "; cin>> Newnode->x;
        cout << "Enter y cordinate: "; cin>> Newnode->y;

        if (Head==NULL)
        {
            //intializing the node
            Head=Newnode;
            Newnode->next=Head;
            Newnode->prev=Head;

        }
        else
        {

            //intializing the node for second iteration

            Newnode-> prev = Head -> prev;
            Head -> prev -> next = Newnode;
            Newnode->next=Head;
            Head -> prev = Newnode;
        }
    }
    return Head;

}

//void Readnode()
void display(Point2D *Head, int n)
{
    Point2D *p=Head;
    for(int i=0; i<n;i++)
    {
        cout << "Point at position:"<<i;
        cout<<endl;
        cout << "Current Points: " << endl;
        cout <<"x: " << p->x ;
        cout <<", y: " << p->y << endl;
        cout << "next Points: " << endl;
        cout <<"x: " << p->next->x ;
```

```
        cout <<", y: " << p->next->y << endl;
        cout << "Previous Points: " << endl;
        cout <<"x: " << p->prev->x ;
        cout <<", y: " << p->prev->y << endl;
        p=p->next;
    }


}

void display1(Point2D (*ptr)){
    //     cout << "Point Name: " << (*ptr).name << endl;
    cout <<"x: " << (*ptr).x << endl;
    cout <<"y: " << (*ptr).y << endl;
}

void display1(Point2D);

Point2D BuildPoint(Point2D (*p2)){
    //     cout << "Enter Point name: ";
    //     cin.get(p2->name, 50); cout << endl;
    cout << "Enter x cordinate: ";
    cin >> (p2->x);
    cout << "Enter y cordinate: ";
    cin >> (p2->y); cout << endl;
return (*p2);
}

getPosition(){
    int p;
    cout<<"enter insert location of point in polgon";
    cin>>p;
    return(p);
}

/* Point2D* InsertAt( Point2D *r, int c)
{
    r->prev=NULL;
    r->next=NULL;

    for(int i=0; i<c;i++)
    {

        if (Head==NULL)
         {
```

```
            //intializing the node
            Head=r;
            Newnode->next=Head;
            Newnode->prev=Head;

        }
        else
        {

            //intializing the node for second iteration

            r-> next = Head;
            Head -> prev= r;
            Head = r;
        }
    }
    return Head;

}

Point2D*  DeleteAt( Point2D *r, int c){
    delete (*r);
};
*/
```

```
int main()
{
    int n,c;
    cout<<"Enter the no of points: ";
    cin>>n;
    Point2D *Head;
    Head = BuildPolygon(n);
    display(Head,3);

    Point2D *ptr2, p2,*r;
    ptr2 = &p2;
    //Reading Point2D Information by pointer2----------------
    cout << "\nReading of Point(BuildPoint()) by pointer2." << endl;
    *r=BuildPoint(ptr2);
    cout << endl;
    //display point2d by pointer2----------------------
    cout << "Displaying point info by pointer2. " << endl;
    display1(ptr2);
    //cout << (*ptr).name << "  " << (*ptr).x << " "<< (*ptr).y << " ";
    cout << endl;
    //enter point location to be inserted
```

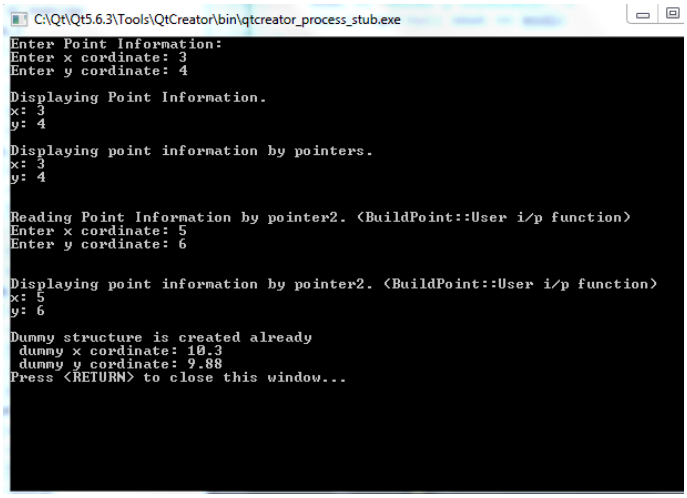

Figure 1: All outputs

```
c=   getPosition();
   /*  InsertAt(*r,c);
     //enter point location to be inserted
     DeleteAt(c);
  */
}
```

****

4 Outputs of all the Point2D

5 Outputs of all the Polygon2D

6 Outputs of insert and delete of Point2D in Polygon2D

6.1 Outputs of insert and delete are not displaying here

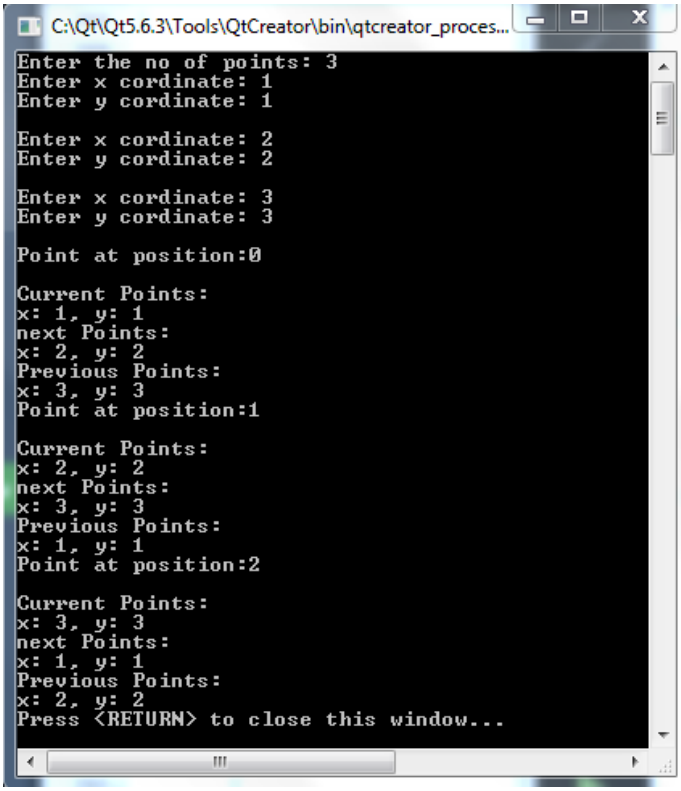

Figure 2: All outputs

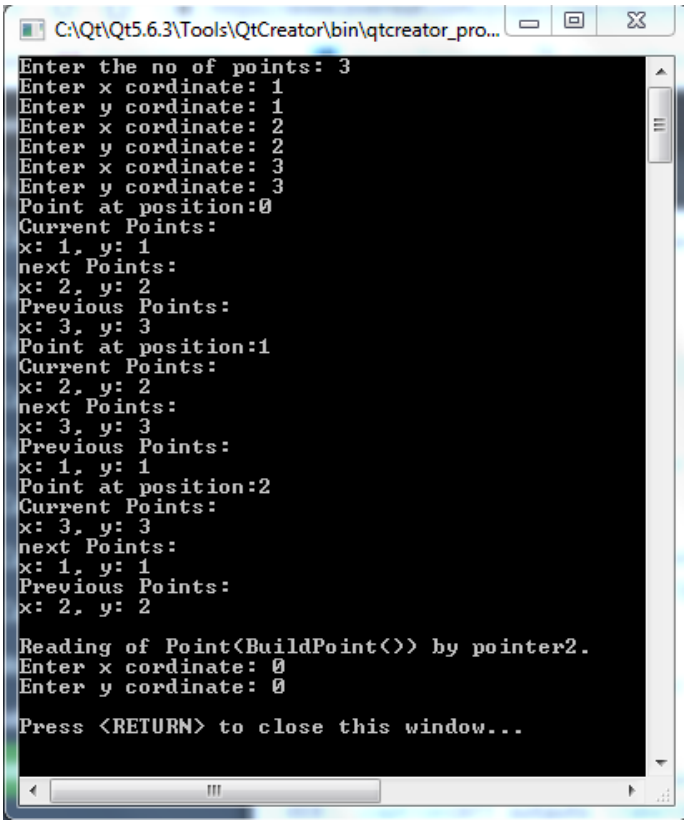

Figure 3: All outputs