Assignment -4 Roll no-20BCS142

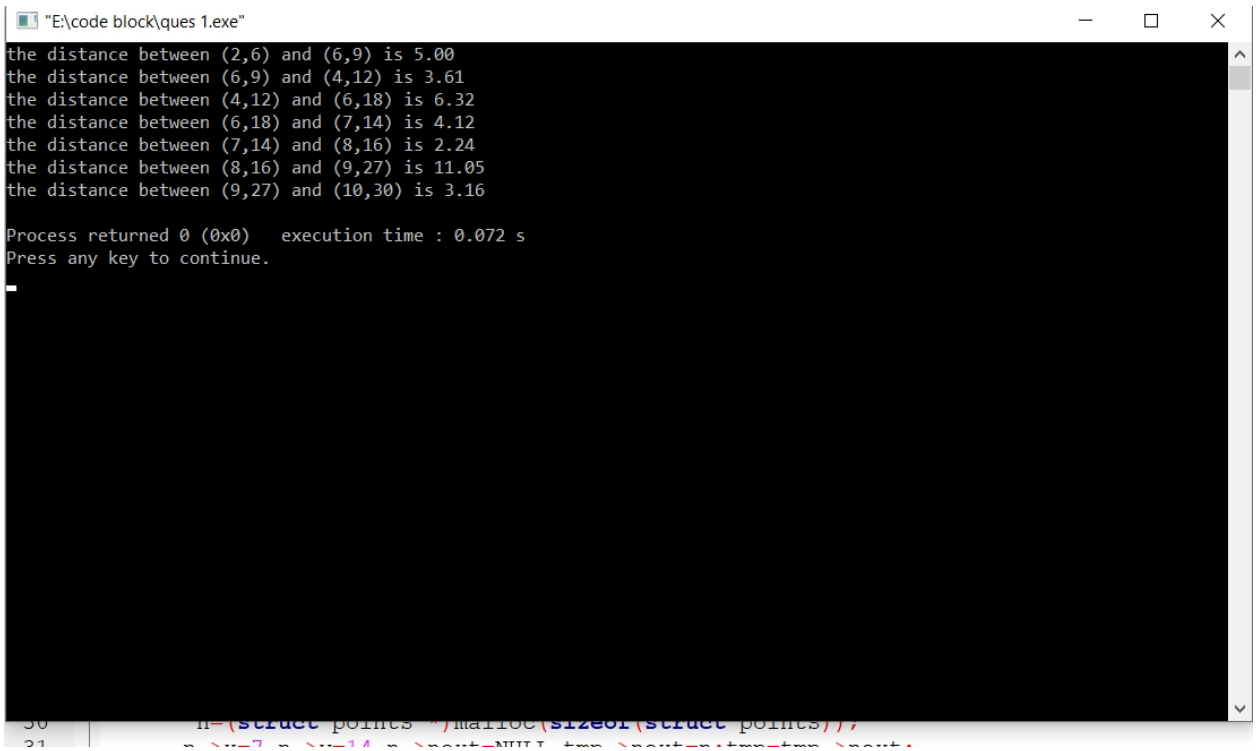
Question no.1

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
struct points{
  int x;
  int y;
  struct points *next;
};
int main() {
struct points *s,*tmp,*n;
  s=(struct points *)malloc(sizeof(struct points));
  s->x=2,s->y=6,s->next=NULL;
 tmp=s;
 int j=2;
/*for(int i=2;i<=10;i++){
      n=(struct points *)malloc(sizeof(struct points));
      n->x=i+1;
      n->y=i*j;
      n->next=NULL;
      tmp->next=n;
```

```
    tmp=n;
    j++;
} */
 n=(struct points *)malloc(sizeof(struct points));
    n->x=6,n->y=9,n->next=NULL,tmp->next=n;tmp=tmp->next;
     n=(struct points *)malloc(sizeof(struct points));
    n->x=4,n->y=12,n->next=NULL,tmp->next=n;tmp=tmp->next;
     n=(struct points *)malloc(sizeof(struct points));
    n->x=6,n->y=18,n->next=NULL,tmp->next=n;tmp=tmp->next;
     n=(struct points *)malloc(sizeof(struct points));
    n->x=7,n->y=14,n->next=NULL,tmp->next=n;tmp=tmp->next;
     n=(struct points *)malloc(sizeof(struct points));
    n->x=8,n->y=16,n->next=NULL,tmp->next=n;tmp=tmp->next;
     n=(struct points *)malloc(sizeof(struct points));
    n->x=9,n->y=27,n->next=NULL,tmp->next=n;tmp=tmp->next;
     n=(struct points *)malloc(sizeof(struct points));
    n->x=10,n->y=30,n->next=NULL,tmp->next=n;tmp=tmp->next;
struct points *ptr;
ptr=s;
while(ptr->next!=NULL){
   float X=(ptr->x -ptr->next->x);
    float Y=(ptr->y -ptr->next->y);
```

```
    float sum=sqrt(X*X + Y*Y);
    printf("the distance between (%d,%d) and (%d,%d) is %.2f\n",ptr->x,ptr->y,ptr->next->x,ptr->next->y,sum);
    ptr=ptr->next;
}
free (s);
free (n);
return 0;
}
```

Output of 1st



Question 2nd

```
#include<stdio.h>
#include<stdlib.h>
struct digit{
int x;
struct digit *next;
};
void Swap(int *x,int *y){
  int t=*x;
  *x=*y;
  *y=t;
}
void permutation(int arr[],int l,int r){
  if(l==r){
   for(int i=0;i<4;i++)
       printf("%d",arr[i]);
   printf("\n");
  }
  else{
       for(int i=l;i<=r;i++){
        Swap(&arr[i],&arr[l]);
        permutation(arr,l+1,r) ;
```

```
        Swap(&arr[i],&arr[l]);
       }


   }
}
int main() {
    struct digit *N,*p;
    int  n,sum=0;
    printf("Enter 4 digit number");
     scanf("%d",&n);
     int arr[5]={0},i=0;
     N=(struct digit *)malloc(sizeof(struct digit));
       N->x=n%10,arr[i++]=n%10;
       N->next=NULL;
       n=n/10;
     while(n){
       p=(struct digit *)malloc(sizeof(struct digit));
       p->x=n%10,arr[i++]=n%10;
       p->next=N;//As we have to store the  address
       N=p;    //such that 1000th digit ->100th digit and
      n=n/10;   //100th digit ->10th digit
     }
```

```
 permutation(arr,0,3);
 while(p!=NULL){
  sum+=((p->x )* (p->x));
  p=p->next;
 }
 printf("sum of Square of digit in the no. is %d",sum);
 free(N);
 free(p);
return 0;
}
```

Output of 2nd