

```

GNU nano 6.2
#include <stdio.h>
#include <unistd.h>
#include <math.h>
void insertionSort(int arr[],int n){
    int i,key,j;
    for(i=1;i<n;i++){
        key = arr[i];
        j=i-1;
        while(j>=0 && arr[j]>key){
            arr[j+1] = arr[j];
            j=j-1;
        }
        arr[j+1]=key;
    }
}

void selectionSort(int arr[],int n){
    int i,j,small;
    for(i=0;i<n-1;i++){
        small = i;
        for(j=i+1;j<n;j++){
            if (arr[j]<arr[small]){
                small=j;
            }
        }
        int temp = arr[small];
        arr[small] = arr[i];
        arr[i] = temp;
    }
}

```

PC1 Desktop environment showing a Command Prompt window. The window displays the results of a ping command to 192.168.10.98 and a subsequent traceroute command. The ping command shows a 25% loss of packets, while the traceroute command shows a successful path to 192.168.10.130.

```

C:\>ping 192.168.10.98

Pinging 192.168.10.98 with 32 bytes of data:

Request timed out.
Reply from 192.168.10.98: bytes=32 time=1ms TTL=126
Reply from 192.168.10.98: bytes=32 time=3ms TTL=126
Reply from 192.168.10.98: bytes=32 time=3ms TTL=126

Ping statistics for 192.168.10.98:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 3ms, Average = 2ms

C:\>ping 192.168.10.98

Pinging 192.168.10.98 with 32 bytes of data:

Reply from 192.168.10.98: bytes=32 time=8ms TTL=126
Reply from 192.168.10.98: bytes=32 time=1ms TTL=126
Reply from 192.168.10.98: bytes=32 time=1ms TTL=126
Reply from 192.168.10.98: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.10.98:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 8ms, Average = 2ms

C:\>D

```

PC0 Desktop environment showing a Command Prompt window. The window displays the results of a traceroute command to 192.168.10.130. The output shows the path taken by the packets, including the source IP, destination IP, and the time taken for each hop.

```

Packet Tracer PC Command Line 1.0
C:\>tracert 192.168.10.130

Tracing route to 192.168.10.130 over a maximum of 30 hops:

  0  0 ms    0 ms    0 ms    192.168.10.1
  1  2 ms    13 ms   0 ms    192.168.10.66
  2  5 ms    1 ms    4 ms    192.168.10.130
  3  *       15 ms   0 ms    192.168.10.130

Trace complete.

C:\>

```

```

void printArray(int arr[],int n){
    int i;
    for(i=0;i<n;i++){printf("%d ",arr[i]);}
    printf("\n");
}

int main(){
    int arr[] = {1,5,6,3,10,11};
    int pid;
    int n= sizeof(arr)/ sizeof(arr[0]);
    pid = fork();
    if(pid==0){
        selectionSort(arr,n);
        printf("SelectionSort : ");
        printArray(arr,n);
    }
    else{
        sleep(1);
        insertionSort(arr,n);
        printf("insertion sort : ");
        printArray(arr,n);
    }
}

```

```

pardha@pardha-virtual-machine:~$ nano test.c
pardha@pardha-virtual-machine:~$ gcc test.c
pardha@pardha-virtual-machine:~$ ./a.out
SelectionSort : 1 3 5 6 10 11
insertion sort : 1 3 5 6 10 11
pardha@pardha-virtual-machine:~$

```