

**Ex No :14(a)**

### **IMPLEMENTATION OF FCFS DISK SCHEDULING ALGORITHM**

**AIM:**

To write a C program for implementation of FCFS Disk Scheduling Algorithm.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Let Request array represents an array storing indexes of tracks that have been requested in ascending order of their time of arrival. 'head' is the position of disk head.

**Step 3:** Let us one by one take the tracks in default order and calculate the absolute distance of the track from the head.

**Step 4:** Increment the total seek count with this distance.

**Step 5:** Currently serviced track position now becomes the new head position.

**Step 6:** Go to step 3 until all tracks in request array have not been serviced.

**Step 7:** Stop the program.

**PROGRAM:**

```
#include<stdio.h>
int main()
{
    int queue[20],n,head,i,j,k,seek=0,max,diff;
    float avg;
    printf("Enter the max range of disk\n");
    scanf("%d",&max);
    printf("Enter the size of queue request\n");
    scanf("%d",&n);
    printf("Enter the queue of disk positions to be read\n");
    for(i=1;i<=n;i++)
        scanf("%d",&queue[i]);
    printf("Enter the initial head position\n");
    scanf("%d",&head);
    queue[0]=head;
    for(j=0;j<=n-1;j++)
    {
        diff=abs(queue[j+1]-queue[j]);
        seek+=diff;
        printf("Disk head moves from %d to %d with seek %d n",
               queue[j],queue[j+1],diff);
    }
    printf("Total Seek Time is %d\n",seek);
    avg=seek/(float)n;
    printf("Average Seek Time is %f\n",avg);
    return 0;
}
```

**OUTPUT:**

**RESULT:**

**Ex No :14(b)**

### **IMPLEMENTATION OF SSTF DISK SCHEDULING ALGORITHM**

**AIM:**

To write a C program for implementation of SSTF (Short Seek Time First) Disk Scheduling Algorithm.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Let Request array represents an array storing indexes of tracks that have been requested. 'head' is the position of disk head.

**Step 3:** Find the positive distance of all tracks in the request array from head.

**Step 4:** Find a track from requested array which has not been accessed/serviced yet and has minimum distance from head.

**Step 5:** Increment the total seek count with this distance.

**Step 6:** Currently serviced track position now becomes the new head position.

**Step 7:** Go to step 3 until all tracks in request array have not been serviced.

**Step 8:** Stop the program.

**PROGRAM:**

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int RQ[100],i,n,TotalHeadMoment=0,initial,count=0;
    printf("Enter the number of Requests\n");
    scanf("%d",&n);
    printf("Enter the Requests sequence\n");
    for(i=0;i<n;i++)
        scanf("%d",&RQ[i]);
    printf("Enter initial head position\n");
    scanf("%d",&initial);
    while(count!=n)
    {
        int min=1000,d,index;
        for(i=0;i<n;i++)
        {
            d=abs(RQ[i]-initial);
            if(min>d)
            {
                min=d;
                index=i;
            }
        }
        TotalHeadMoment=TotalHeadMoment+min;
        initial=RQ[index];
        RQ[index]=1000;
```

```
        count++;  
    }  
    printf("Total head movement is %d",TotalHeadMoment);  
    return 0;  
}
```

**OUTPUT:**

**RESULT:**

**Ex No :14(c)**

### **IMPLEMENTATION OF SCAN DISK SCHEDULING ALGORITHM**

**AIM:**

To write a C program for implementation of SCAN Disk Scheduling Algorithm.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Let Request array represents an array storing indexes of tracks that have been requested in ascending order of their time of arrival. 'head' is the position of disk head.

**Step 3:** Let direction represents whether the head is moving towards left or right.

**Step 4:** In the direction in which head is moving service all tracks one by one.

**Step 5:** Calculate the absolute distance of the track from the head.

**Step 6:** Increment the total seek count with this distance.

**Step 7:** Currently serviced track position now becomes the new head position.

**Step 8:** Go to step 4 until we reach at one of the ends of the disk.

**Step 9:** If we reach at the end of the disk reverse the direction and go to step 3 until all tracks in request array have not been serviced.

**Step 10:** Stop the program.

**PROGRAM:**

```
#include<stdio.h>
main()
{
    int t[20], d[20], h, i, j, n, temp, k, atr[20], tot, p, sum=0;
    printf("Enter the no of tracks to be Traversed : ");
    scanf("%d",&n);
    printf("\nEnter the position of Head: ");
    scanf("%d",&h);
    t[0]=0;
    t[1]=h;
    printf("\nEnter the Tracks: ");
    for(i=2;i<n+2;i++)
        scanf("%d",&t[i]);
    for(i=0;i<n+2;i++)
    {
        for(j=0;j<(n+2)-i-1;j++)
        {
            if(t[j]>t[j+1])
            {
                temp=t[j];
                t[j]=t[j+1];
                t[j+1]=temp;
            }
        }
    }
    for(i=0;i<n+2;i++)if(t[i]==h)
```

```

j=i;k=i;
p=0;
while(t[j]!=0)
{
    atr[p]=t[j];
    j--;
    p++;
}
atr[p]=t[j];
for(p=k+1;p<n+2;p++,k++)
    atr[p]=t[k+1];
for(j=0;j<n+1;j++)
{
    if(atr[j]>atr[j+1])
        d[j]=atr[j]-atr[j+1];
    else
        d[j]=atr[j+1]-atr[j];
    sum+=d[j];
}
printf("\nAverage Header Movements:%f \n", (float)sum/n);
}

```

**OUTPUT:**

**RESULT:**

**Ex No :14(d)**

### **IMPLEMENTATION OF C-SCAN DISK SCHEDULING ALGORITHM**

**AIM:**

To write a C program for implementation of C-SCAN Disk Scheduling Algorithm.

**ALGORITHM:**

**Step 1:** Start the program.

**Step 2:** Let Request array represents an array storing indexes of tracks that have been requested in ascending order of their time of arrival. 'head' is the position of disk head.

**Step 3:** The head services only in the right direction from 0 to the size of the disk.

**Step 4:** While moving in the left direction do not service any of the tracks.

**Step 5:** When we reach the beginning(left end) reverse the direction.

**Step 6:** While moving in the right direction it services all tracks one by one.

**Step 7:** While moving in the right direction calculate the absolute distance of the track from the head.

**Step 8:** Increment the total seek count with this distance.

**Step 9:** Currently serviced track position now becomes the new head position.

**Step 10:** Go to step 6 until we reach the right end of the disk.

**Step 11:** If we reach the right end of the disk reverse the direction and go to step 4 until all tracks in the request array have not been serviced.

**Step 12:** Stop the program.

**PROGRAM:**

```
#include<stdio.h>
main()
{
    int t[20], d[20], h, i, j, n, temp, k, atr[20], tot, p, sum=0;
    printf("\nEnter the no of tracks to be Traversed: ");
    scanf("%d",&n);
    printf("\nEnter the position of head: ");
    scanf("%d",&h);
    t[0]=0;
    t[1]=h;
    printf("\nEnter total tracks: ");
    scanf("%d",&tot);
    t[2]=tot-1;
    printf("\nEnter the tracks: ");
    for(i=3;i<=n+2;i++)
        scanf("%d",&t[i]);
    for(i=0;i<=n+2;i++)
        for(j=0;j<=(n+2)-i-1;j++)
            if(t[j]>t[j+1])
            {
                temp=t[j];
                t[j]=t[j+1];
```

```

                                t[j+1]=temp;
                                }
for(i=0;i<=n+2;i++)
if(t[i]==h)
{
    j=i;
    break;
}
p=0;
while(t[j]!=tot-1)
{
    atr[p]=t[j];
    j++;
    p++;
}
atr[p]=t[j];
p++;
i=0;
while(p!=(n+3) && t[i]!=t[h])
{
    atr[p]=t[i];
    i++;
    p++;
}
for(j=0;j<n+2;j++)
{
if(atr[j]>atr[j+1])
    d[j]=atr[j]-atr[j+1];
else
    d[j]=atr[j+1]-atr[j];
    sum+=d[j];
}
printf("\nTotal header movements: %d",sum);
printf("\nAvg is: %f",(float)sum/n);
}

```

**OUTPUT:**

**RESULT:**



**Ex No :15**

## **INSTALLATION OF WINDOWS OPERATING SYSTEM**

**AIM:**

To install windows 10 operating system.

**PROCEDURE:**

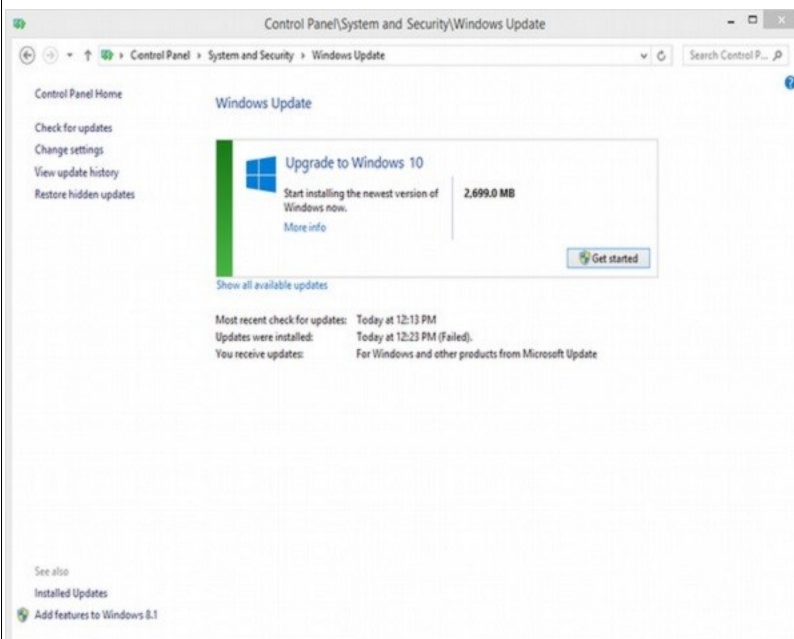
**Step 1 :**

Look for the Windows 10 notification in the lower-right corner of the screen. This is a one-year-only offer that Microsoft is extending to valid users of Windows 7 and Windows 8.1.



**Step 2 :**

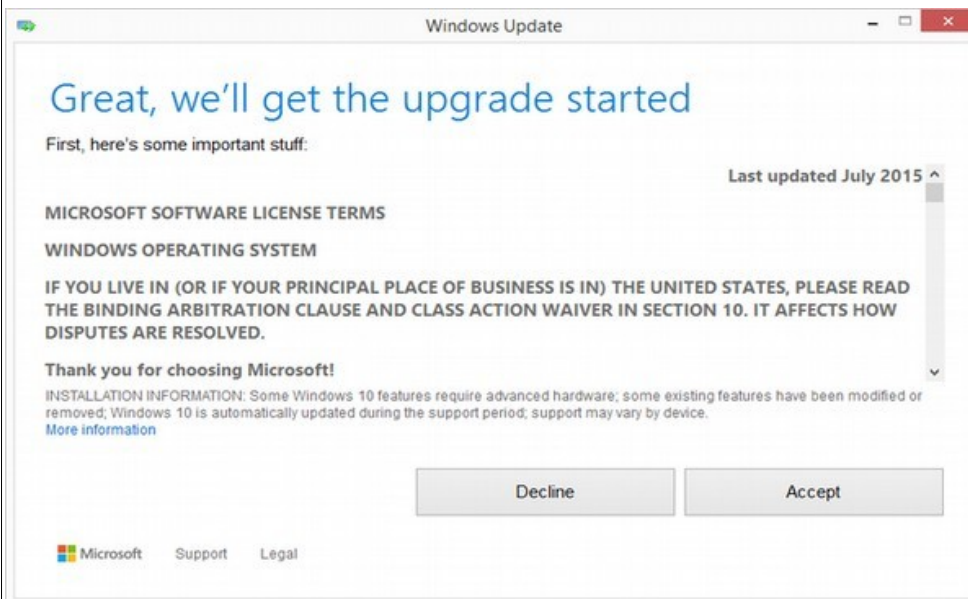
By clicking on the notification, it will start the download and installation process of Windows 10 in your system.



The download required for the upgrade is quite large, so make sure you have a stable Internet connection and continuous power for your computer to avoid interruptions during the process.

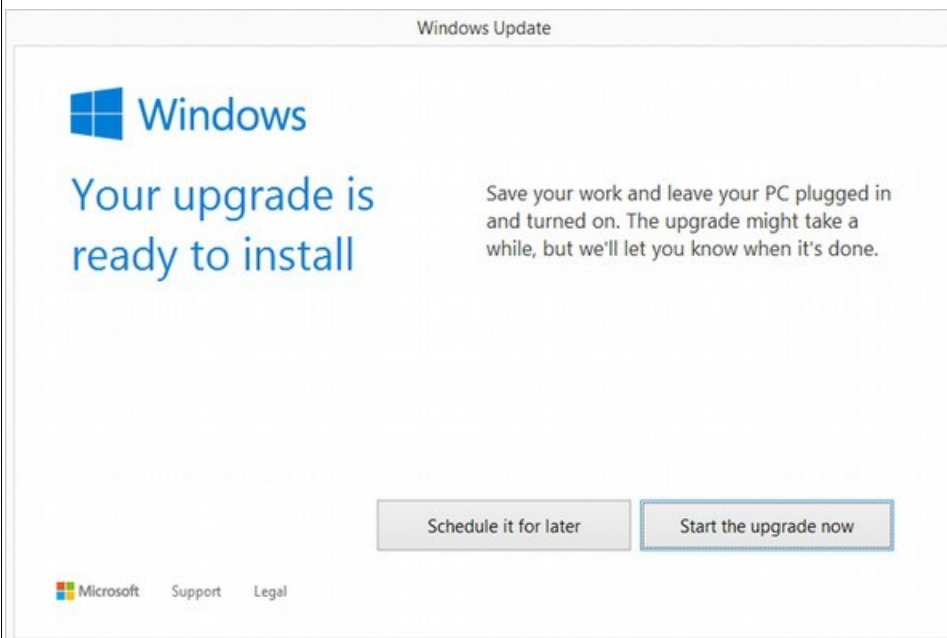
### Step 3 :

After the download is complete, it will prompt you to accept Microsoft's license terms.



### Step 4:

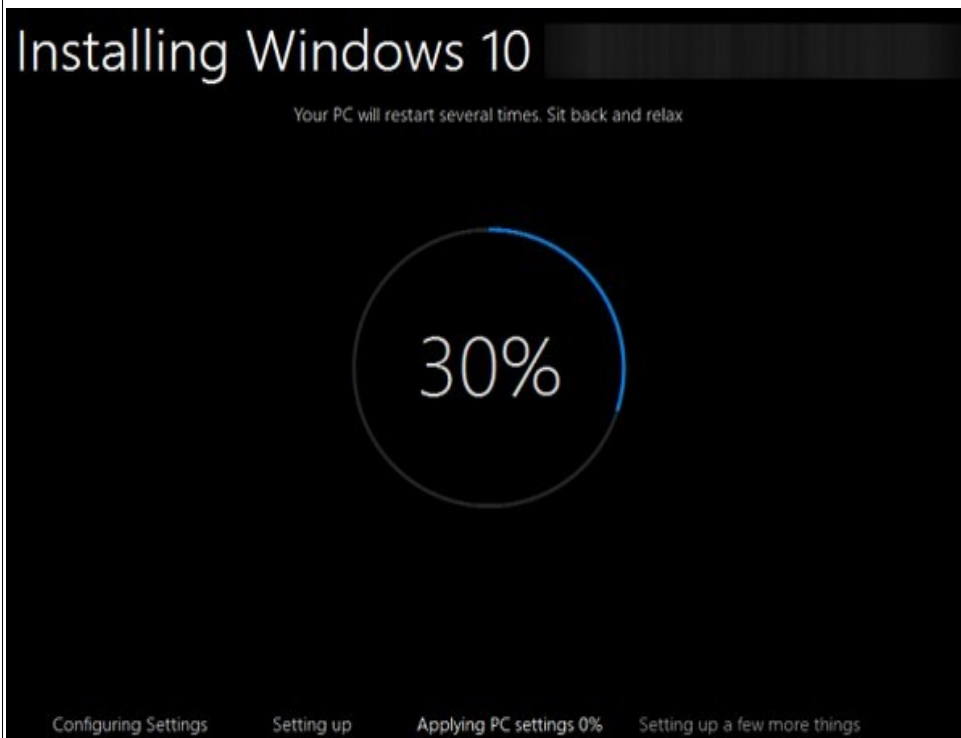
After agreeing to the terms, it will ask if you want to install the upgrade at that moment or schedule it for later.



Since the upgrade process can take approximately 2 hours, it will be helpful to schedule it for a time, that will be more suitable to you.

### Step 5:

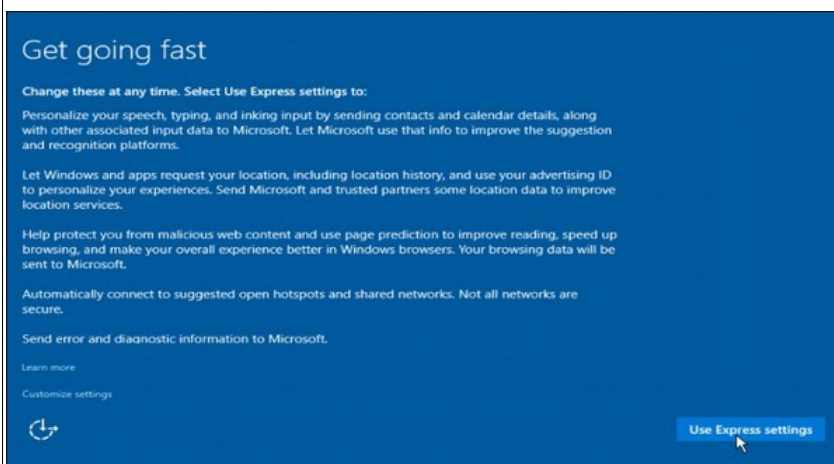
Once the upgrade starts, the system will perform a series of tasks, during which you will see the following screen.



During this time, your computer will reboot a couple of times, so don't worry. The process itself will take you through the steps to complete the upgrade.

### Step 6:

As the upgrade approaches its end, it will ask you to configure some basic Windows settings. You can choose to use **Express settings**, which will apply the most common or recommended settings, or you can choose to **customize the settings** as you please.



## Step 7:

After the upgrade finishes, you'll see the Windows welcome screen.

### Clean Install

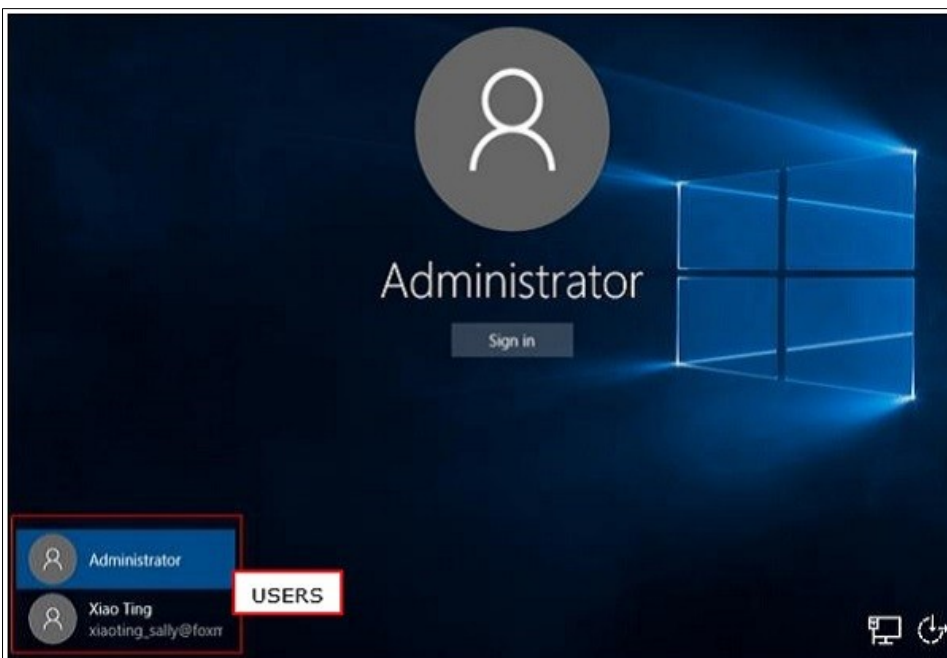
If your computer has an older operating system like Windows XP or Vista, you won't be able to upgrade for free. In these cases, you'll need to buy a boxed copy of Windows 10 to have a valid license for the installation. Windows 10 will have a starting price of \$119.



But take in consideration that old computers that have either XP or Vista installed might have obsolete hardware components and might not be suitable for Windows 10. In this case, make sure you review the system requirements listed at the beginning of this chapter to check if your computer is qualified for a Windows 10 upgrade.

If you choose this type of an installation, just insert the disc in your computer and turn it on. Most computers will ask you to press a specific key to boot from the CD/DVD, but most systems use the F12 key. After accessing the disc, you just have to follow the steps which are very similar to the ones from the upgrade.

After you have installed or upgraded your Windows, you will get a Welcome Screen with the time and date. Just click anywhere to go to the User Accounts Screen.



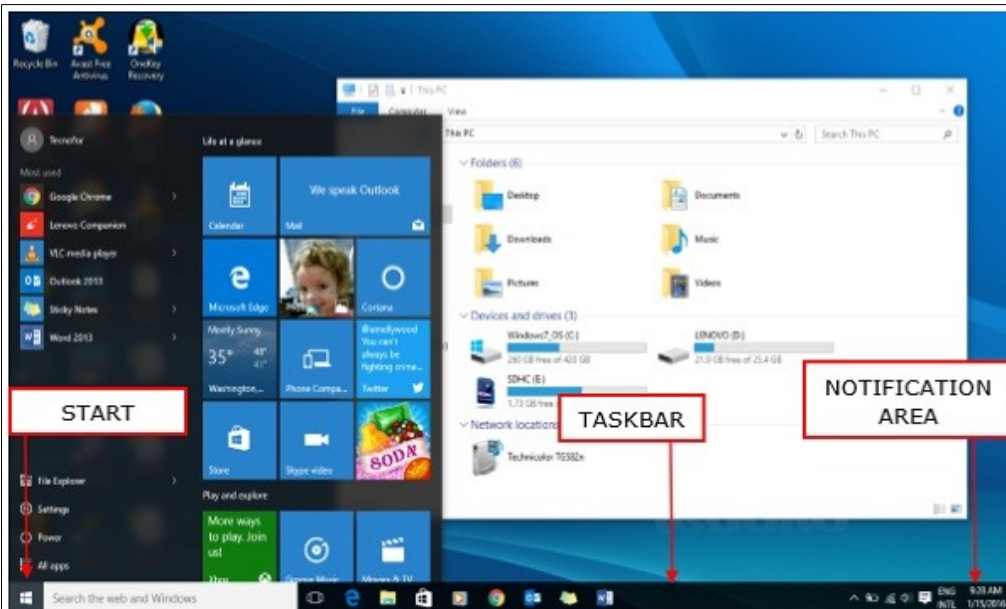
This screen lets you choose which user you want to log in to from the lower-left corner. After choosing the right user, and entering a password if necessary, you will see the Windows Desktop.



The Windows Desktop is simply your operating system main screen. Here you have access to an array of tools like the Start Menu, Taskbar, and other icons.

Windows 10 also introduces a search box in the Taskbar, which facilitates browsing both your computer and the Web.

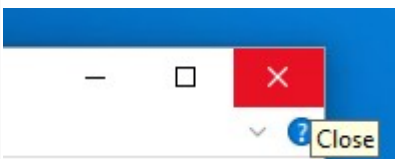
Once you get to the Windows Desktop screen, here are some basic features you will see.



One of the most important parts of your Desktop is the Taskbar. By default, it sits at the bottom of your screen giving you access to the Start Menu, several application icons, and the Notification Area.

## Windows

In Windows 10, if an application is active or opened, you will see a green line below its icon. Clicking the icon will bring the application window up.



Every open window features three buttons in the upper-right corner. These are used to minimize, maximize, or close the window –

- Minimizing means that the window will hide in the Taskbar.
- Maximizing will bring the window to a full-screen size.

Windows can be moved around or resized as you please –

- To move a window, just click on its Title Bar on the upper side of the window and drag it.
- To resize a window, move your mouse to any corner until you see a double-sided arrow. Then click and drag until you reach the desired size.

## RESULT:

Thus the windows 10 operating system was installed successfully.



**Ex No :16**

## **INSTALLATION OF GUEST OPERATING SYSTEM USING VMWARE**

**AIM:**

To install the guest operating system using vmware.

**PROCEDURE:**

**Step 1:**

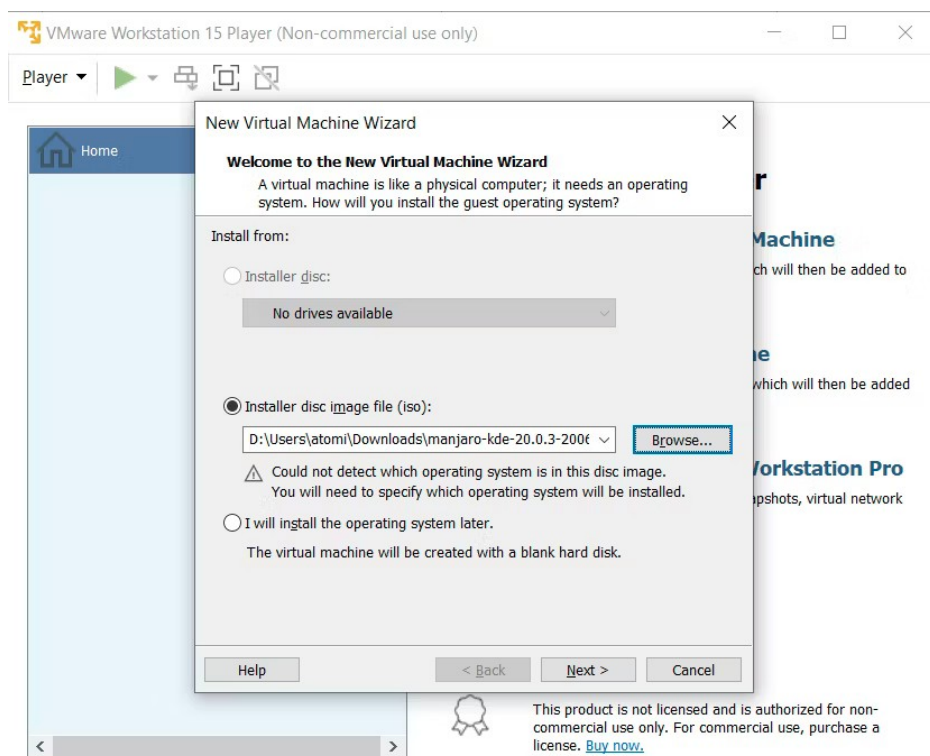
Click Create a New Virtual Machine

**Step 2:**

Select the default option, **Installer disc image file (iso)**

**Step 3:**

Click **Browse** to find the ISO file

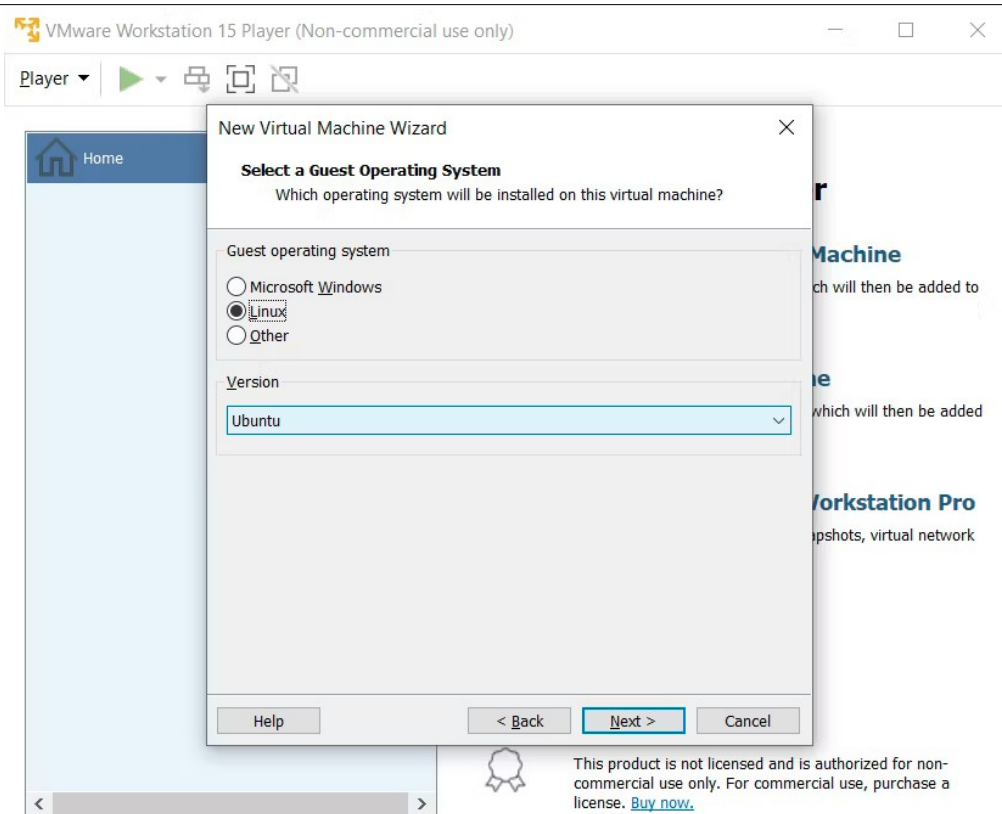


**Step 4:**

With "guest" OS selected, click **Next**

**Step 5:**

Select **Linux** as the Guest operating system type



**Step 6:**

Under **Version**, scroll through the list and select the OS.

**Step 7:**

Click **Next** to proceed and if necessary, input a **Virtual machine name**

**Step 8:**

Confirm the storage **Location** and change if needed

**With the operating system selected and configured, it's time to build the virtual machine.**

**Step 9:**

Under **Specify Disk Capacity** adjust **Maximum disk size** if required (the default should be enough)

**Step 10:**

Select **Split virtual disk into multiple files** as this makes moving the VM to a new PC easy

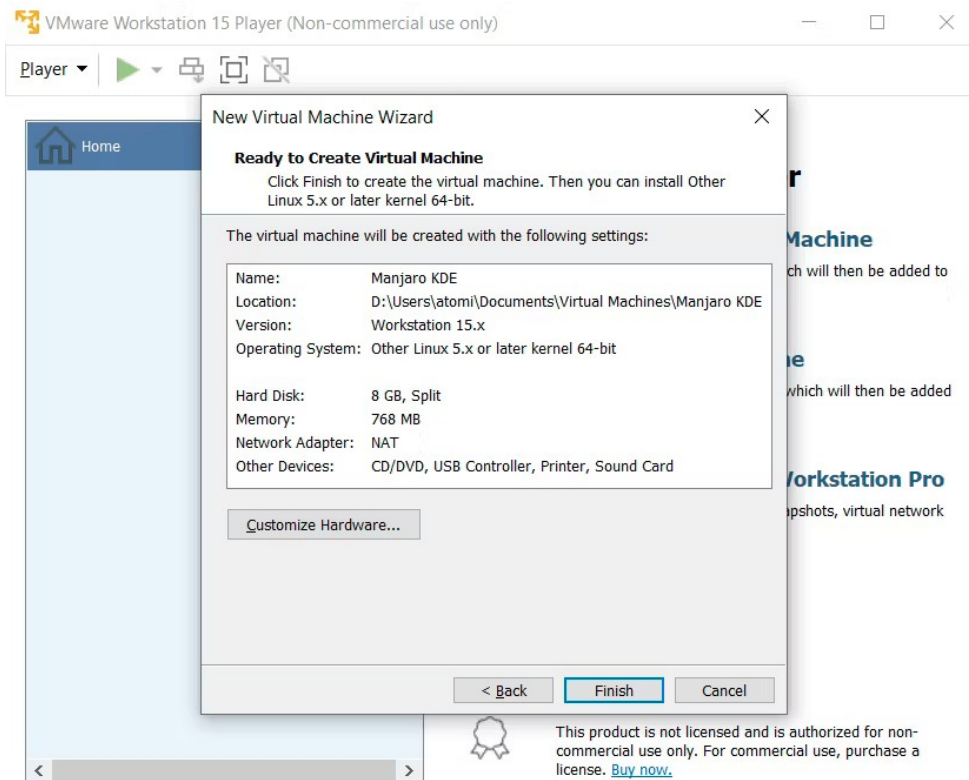
**Step 11:**

Click **Next** then confirm the details on the next screen

**Step 12:**

If anything seems wrong click **Back**, otherwise click **Finish**



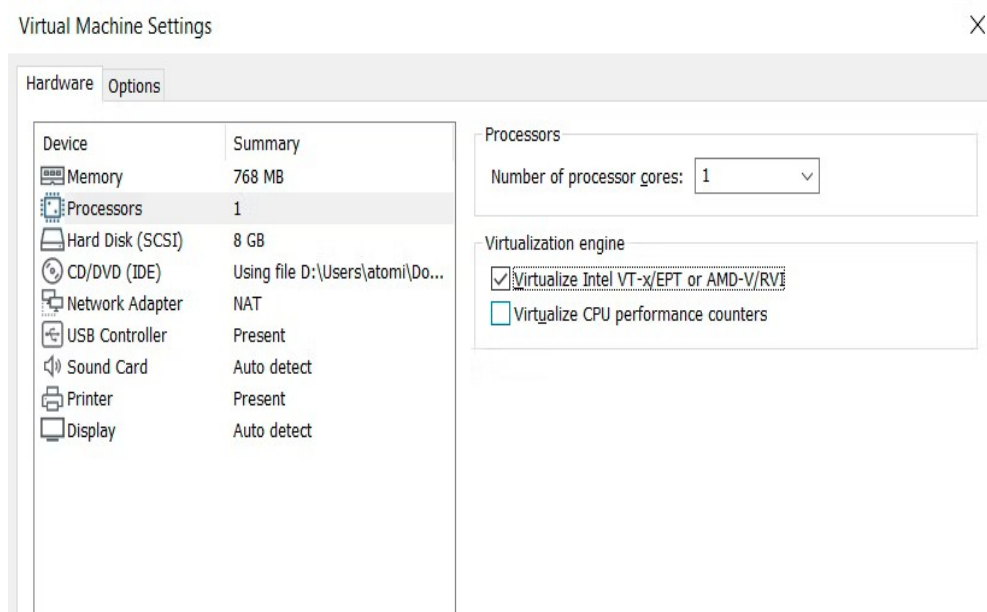


Your Linux virtual machine will be added to VMware Workstation Player.

## Customize Your Virtual Hardware

In some cases, you might need to customize the virtual machine before installing Linux. Alternatively, you might install the OS and find there is something missing.

To fix this, right-click your virtual machine in VMware Workstation Player and select **Settings**.



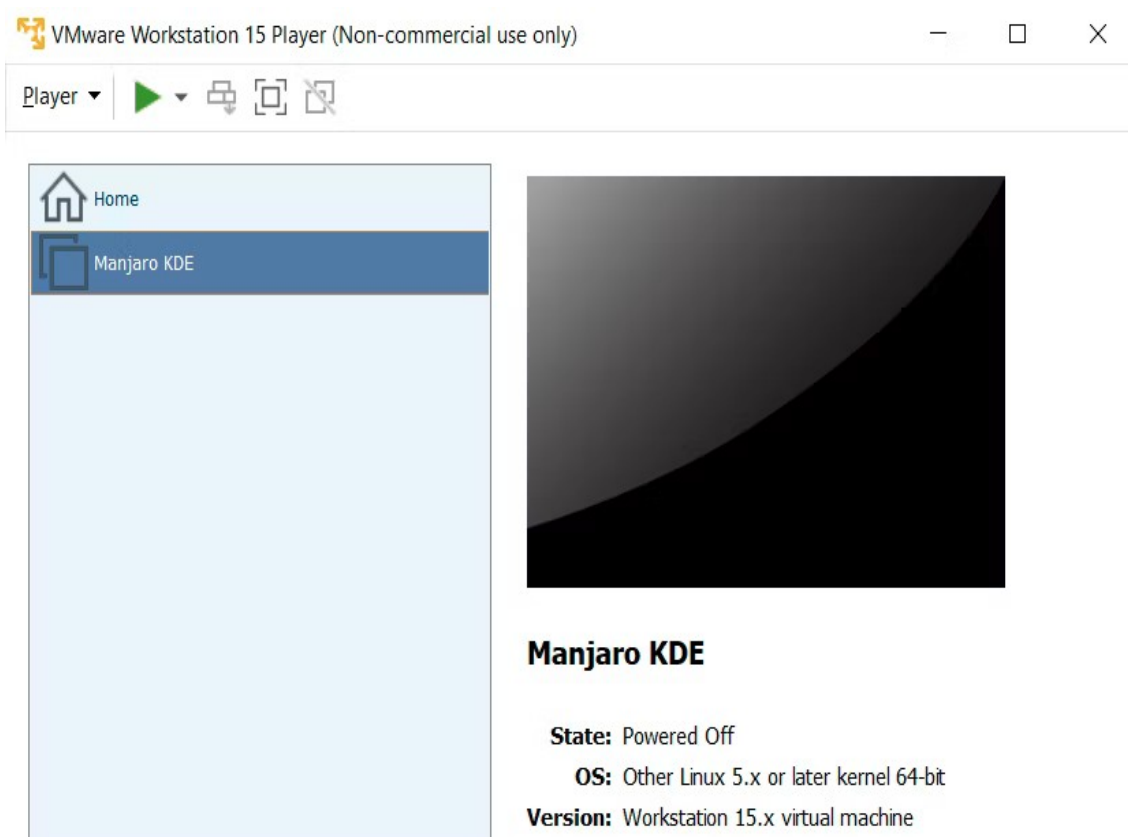
Here, you can tweak the virtual machine's hardware in other ways beyond the HDD. You have options for the **Memory**, **Processors**, **Network Adaptor** configuration, and much more.

It's worth taking a look at the **Processors** screen. In the right-hand pane, you'll spot a reference to a **Virtualization engine**. By default, this works automatically, but for troubleshooting set Intel VT-x or AMD-V, depending on your CPU.

You can address performance issues in the **Memory** screen. Here you'll spot an illustration of the suggested RAM size, as well as recommended options for your virtual machine. It's a good idea to stick to these recommendations. Going too small will prove a problem, while setting the RAM too high will impact on your PC's performance, slowing everything from standard system tasks to running the VM software!

Finally, spare a moment to check the **Display** settings. Default settings should be fine but if there is an issue with the display you can toggle 3D acceleration. Multiple monitors can be used and custom resolution set, but note that some modes will clash with some desktops.

Click **OK** to confirm changes, then select the virtual machine and click the **Play** button to begin.



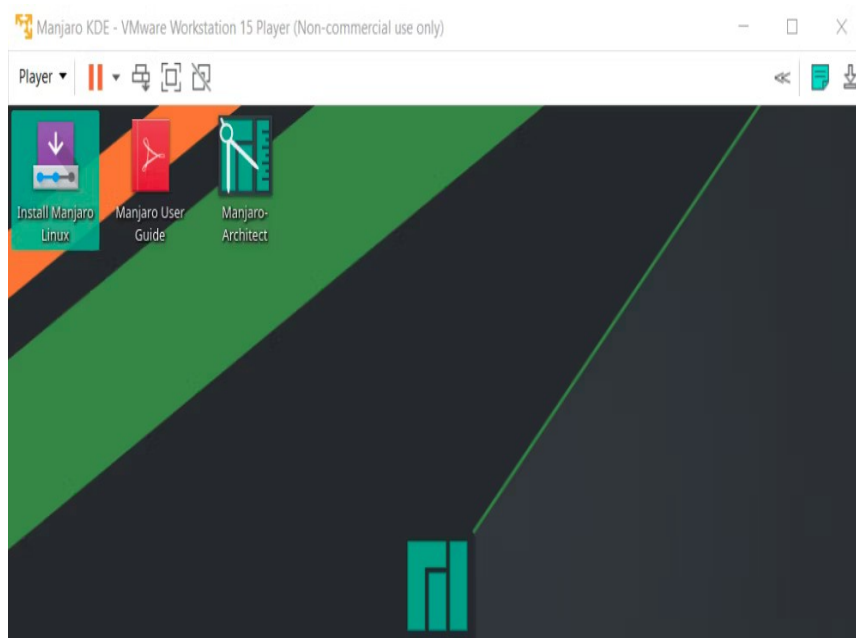
## Download and Install VMware Tools

On the first boot of your virtual machine, you'll be prompted to **Download and Install** VMware Tools for Linux. Agree to this, then wait as it is downloaded.

VMware Tools will enhance the performance of the virtual machine while enabling shared folders between host and guest machines.

## How to Install Linux in VMware

When the ISO boots in the virtual machine, it will boot into the live environment. This is a temporary Linux that exists only on the boot media and in the system memory. To ensure the environment persists, use the **Install** option on the desktop.



## RESULT:

Thus the the guest operating system using vmware was installed successfully.