CHAPTER – 5

WINDOWS AND UBUNTU

1. An operating system is system software.
2. Operating system enables the hardware to communicate and operate with other software.
3. Operating system acts as an interface between the user and the hardware.
4. Windows series for desktop and laptop computers.
5. Android series for smart phones.
6. Ios for apple phones, i-pod and i-pad.
7. Linux is the open source operating system for desktop and server.
8. Microsoft windows is one of the most popular GUI.
9. Multiple applications can execute simultaneously in windows, and this is known as multitasking.
10. Mouse is used to interact with windows by clicking its icons.
11. Keyboard is used to enter alphabets, numerals and special characters.
12. In which versions of windows mouse was introduced windows 1.x
13. Windows 3.x introduced the concept of multitasking.
14. In which year mouse was introduced as an input device 1985
15. In which versions windows introduced start button, and start menu windows95
16. Plug and play feature was introduced in windows98
17. Version for designed to act as server in network is windows NT
18. Four versions of windows 2000 were released.
19. Professional for windows desktop and laptop systems.
20. Server for both a web server and an office server.
21. Advanced server for line of business applications.
22. Data center server for high-traffic computer network.
23. Start button was removed in windows 8 versions.
24. In windows 10 cortana voice activated personal assistant introduced.
25. Point to the item on the screen then press and release the left mouse button.
26. Clicking on the right mouse button displays a pop up menu.
27. The opening screen of windows called as desktop.

28.Icons is a graphic symbol representing the window elements like files, folders, and shortcut.

29.The default icons are called as standard icons.

30.The disk drive icons options graphically represent in five.

31.The larger window is called applications window.

32.The smaller window which is inside the application window is called document window.

33.We can see two windows on the screen.

34.Title bar will display the name of the application and the name of the document opened.

35.Title bar contain minimize, maximize and close button.

36.Under the title bar it seen menu bar.

37.Which function key focus on the first menu of the menu bar f10.

38.Workspace is the area in the document window to enter the text.

39.The scroll bars are horizontally and vertically bars.

40.To resize the windows which helps to drag and resize in corners and borders.

41.The mouse pointer changes to a double headed arrow when positioned over a border.

42.The lower-left-hand corner of the windows screen is the start button.

43.At the bottom of the screen is a horizontal bar called as task bar.

44.Next to the start button is the quick launch toolbar.

45.The user can see the disk drivers mounted in my computer.

46.Quick launch toolbar which contains task for frequently used application.

47.In windows 8 and 10 my computer called as This PC.

48.Installed applications on your computer are available through the start menu.

49.To quit a application, click the close button in the upper right corner of the application window.

50.Shortcut key for quit the application is alt+f4.

51.Menu for close the application is file->exit or file->close.

52.Open any drive where you want to create a new folder click on file->new->folder.

53.A folder appears with the default name is new folder.

54.WordPad is an in-built word processor application in window OS.

55.Menu used to save a file file->save.
56.Shortcut key used to save a file ctrl+s.
57.Select the location where you want to save the file by using look in drop-down list box.
58.Search box in the start menu to quickly search a particular folder or file.
59.Shortcut key to open a file is ctrl+o.
60.To rename the file, click file->rename.
61.Shortcut key for rename is f2.
62.Shortcut key for cut is ctrl+x.
63.Shortcut key for copy is ctrl+c.
64.Shortcut key for paste is ctrl+v.
65.The menu used for cut is edit->cut.
66.The menu used for copy is edit->copy.
67.The menu used for paste is edit->paste.
68.Release the menu button when the target is highlighted.
69.Types of panes in window are two.
70.When you delete a file or folder, it will move in to the Recycle Bin.
71.To Permanently delete a file or folder hold down the shift+Delete key.
72.Right click on a file or folder to be restored and select Restore option from the pop-up menu.
73.To Delete a file in the Recycle bin, select Empty the Recycle bin from Right click option.
74.To switch to another user account on the computer use Switch user to without closing your open programe.
75.Logoff switch to another user account on the computer after closing all your open program.
76.Lock keys is used to lock the computer.
77.To Reboot the computer use the option Restart.
78.Sleep mode used to put the computer in to a low-power mode.
79.Hibernate put the computer in to a low-power mode after saving an running program.
80.Right click on the file or folder, then select send to for create short cut of file or folder.

81. Open Source refers to a program in which the source code is available in the web.
82. Open Source changes within the community in web.
83. Linux is one of the popular open source in unix.
84. Ubuntu is a linux based operating system.
85. The desktop version supports all normal software like windows in ubuntu.
86. Libre Office is like office of windows.
87. Ubuntu has in-built email software called Thunder bird.
88. Network indicator manages network connection.
89. The current keyboard layout settings shows in Text entry.
90. Toolbar displays your directions browsing history.
91. The default ubuntu 16.04 theme is known as Ambiance.
92. The vertical bar icons on the left side of the desktop is called Launchor.
93. File icon is equivalent to my computer.
94. Trash icon is equivalent to Recycle bin.
95. Libre office calc is the equivalent to spread sheets.

CHAPTER – 6

SPECIFICATION AND ABSTRACTION

96. An algorithm is a sequence of instruction to solve a problem.
97. An algorithm is a sequence of instruction to solve a problem
98. Instructions of a computer are also known as statements.
99. A computer can only execute instructions in a programming language.
100. A program is specified by given input and desired output.
101. Algorithm starts execution with the input, executes the data and gives output.
102. How many steps Mr.polya used to solve the problem four.
103. How many basic building blocks of construct are there four.
104. How many types of data are there two.
105. Variables are named boxes for storing data.
106. The data stored in a variable is also known as the value of variable.
107. Text and numbers are called as data.
108. Algorithm is a sequence of statements.
109. The sequences of statements are executed one after another in the same order is called sequential control flow.
110. There are three important control flow statements.
111. The condition of the state is tested, and if the condition is true, one statement is executed is called as alternative control flow.
112. A condition of the state is tested, and if the condition is true, a statement is executed repeated until the condition is false Is called as iterative control flow.
113. The parts of algorithm are known as functions.
114. Specification is the first step in problem solving is to state the problem precisely.
115. Hiding unnecessary details is known as abstraction.
116. A double dash indicates that the rest of the line is a comment in algorithm.
117. A double slash indicates that the rest of the line is a comment in c++.
118. An assignment statement is used to store a value in variable.

119. Programming language provides basic statements and a notation for comparing compound statements.
120. Specification of an algorithm is a contract between the designer and user of the algorithm.
121. If i=3 before the assignment, i:=i - - after the assignment, the value of I is 2.
122. Assignment statement changes the values of variables, and hence the state.
123. C=sum(a,b) is addition function.
124. A process which accomplishes the intended task.
125. - - inputs : n, n is a real number, n>=0.
- - outputs: y is a real number such that y^2 = n. is a square root(n).
126. Assemble a bicycle is nature activities of algorithm.
127. Multiply two numbers is not a nature of algorithm activities.
128. If 0<I before the assignment i:=i+1 after the assignment we can conclude that : 0>=i
129. The parts of an algorithm are known as functions.
130. Variables are named boxes for storing data.

CHAPTER – 7

COMPOSTION AND DECOMPOSITION

1. There are mainly three different notations for representing algorithms.
2. A programming language is a notation for expressing algorithms to be executed by computers.
3. Pseudo code is a notation similar to programming language.
4. Flow chart is a diagrammatic notation for representing algorithm.
5. An algorithm expressed in a programming language is called as program.
6. Translator which translates the program into instruction executable by the computer.
7. Diamond shaped represent for condition.
8. A statement is contained in a rectangular box.
9. Parallelogram boxes represent inputs and outputs.
10. Control flow statements are compound statements.
11. A special box indicates the start and end of flowchart.
12. There are three important control flow statements.
13. A sequential statement is composed of a sequence of statements.
14. Sequential statement executes one after another.
15. Alternative statement tests the statement and gives the true or false statement.
16. Case analysis statement generalizes it to multiple cases.
17. While statement is iterative statement.
18. Iterative process executes the same repeatedly until the condition false.
19. The false arrow points to the box after the iterative statement.
20. Decomposition is one of the elementary problem – solving techniques.
21. Sub – algorithm is called as function.
22. Alternative statement selects and executes exactly one of the two statements, depending on the value of the condition.
23. A function is an abstraction of a subprogram.
24. The variant of alternative statement is called as condition statement.
25. Connector is used to connecting the flowchart if its length.
26. If a>b, a is true and b is false, the compound statement.

1.if a

2. a statement

3. else

4. if b

5. b statement

6. else

7. end statement

Executes

Answer: a statement

27.If I is false just before the loop, the control flow through: I,a:=1,2

1. S1
2. Do
3. S2
4. While i>a
5. S3

Answer : s1,s2,s3

28.Suppose A, B:= 5,7 before the assignment. What are the values of u and v after the sequence of assignment?

1. C:=B
2. A:=B
3. B:=C

Answer a=7 and b=5

29.How many times the loop is iterated?

1. I:=0
2. While I=5
3. I:=I ++

Answer 6

30.How many times the loop is iterated?

1. I:=10
2. While I<5
3. I:=I - 2

Answer 3

CHAPTER – 8

ITERATION AND RECURSION

1. Recursion must have one base case.
2. Recursion is more powerful than iteration.
3. Recursion used to solve the same problem.
4. The loop invariant is true in four crucial points in the loop.
5. An invariant the loop body is known as a loop invariant.
6. When the solver calls a sub-solver is known as recursive call.
7. At the start of each iteration a loop invariant needs not be true.
8. Recursion is a method call to the same method.
9. Who coined the phrase “structured programming”? Dijkstra
10. Recursion must have at least one base case.
11. Iteration makes the code longer.
12. Recursion reduces the size of the code.
13. Iteration allows the set of instructors to be executed repeatedly.
14. Recursion is always applied to functions.
15. When the loop ends, the termination condition should be established.
16. Certain problems are solved quite easily by recursion.
17. Recursive formula can be used to find the factorial of a number? fact(n) = n * fact(n – 1)
18. Base cases that determine when to stop.

Some examples for recursion algorithm

Example 1: Algorithm for finding the ***k***-th even natural number
Note here that this can be solved very easily by simply outputting $2*(k - 1)$ for a given ***k* .** The purpose here, however, is to illustrate the basic idea of recursion rather than solving the problem.

Algorithm1:Even(positive integer ***k*)**
Input: ***k*** , a positive integer
Output: ***k***-th even natural number (the first even being ***0***)

Algorithm:

```
if k = 1, then return 0;
else return Even(k-1) + 2 .
```

Here the computation of **Even(*k*)** is reduced to that of **Even** for a smaller input value, that is **Even(*k-1*). Even(*k*)** eventually becomes **Even(*1*)** which is ***0*** by the first line. For example, to compute **Even(*3*), Algorithm Even(*k*)** is called with ***k* = *2***. In the computation of **Even(*2*), Algorithm Even(*k*)** is called with ***k* = *1*.** Since **Even(*1*) = *0, 0*** is returned for the computation of **Even(*2*)**, and **Even(*2*)** = **Even(*1*) + *2* = *2*** is obtained. This value ***2*** for **Even(*2*)** is now returned to the computation of **Even(*3*)**, and **Even(*3*) = Even(*2*) + *2* = *4*** is obtained.
As can be seen by comparing this algorithm with the recursive definition of the set of nonnegative even numbers, the first line of the algorithm corresponds to the basis clause of the definition, and the second line corresponds to the inductive clause.

By way of comparison, let us see how the same problem can be solved by an iterative algorithm.

Algorithm 1-a: Even(positive integer ***k*)**
Input: ***k***, a positive integer

Output: ***k***-th even natural number (the first even being ***0***)

Algorithm:

```
int  i, even;
i := 1;
even := 0;
while( i < k ) {
        even := even + 2;
        i := i + 1;
}
return even .
```

Example 2: Algorithm for computing the ***k***-th power of ***2***

Algorithm 2 Power_of_2(natural number **k)**
Input: ***k*** , a natural number
Output: ***k***-th power of ***2***

Algorithm:

```
if k = 0, then return 1;
else return 2*Power_of_2(k - 1) .
```

By way of comparison, let us see how the same problem can be solved by an iterative algorithm.

Algorithm 2-a Power_of_2(natural number **k)**
Input: ***k*** , a natural number
Output: ***k***-th power of ***2***

Algorithm:

```
int  i, power;
i := 0;
```

```
power := 1;
while( i < k ) {
        power := power * 2;
        i := i + 1;
}
return power .
```

The next example does not have any corresponding recursive definition. It shows a recursive way of solving a problem.

Example 3: Recursive Algorithm for Sequential Search

Algorithm 3 SeqSearch(*L, i, j, x*)

Input: ***L*** is an array, ***i*** and ***j*** are positive integers, $i \leq j$, and ***x*** is the key to be searched for in ***L***.

Output: If ***x*** is in ***L*** between indexes ***i*** and ***j***, then output its index, else output ***0***.

Algorithm:

```
if i ≤ j , then
{
  if L(i) = x, then return i ;
  else return SeqSearch(L, i+1, j, x)
}
else return 0.
```

Recursive algorithms can also be used to test objects for membership in a set.

Example 4: Algorithm for testing whether or not a number ***x*** is a natural number

Algorithm 4 Natural(a number ***x*****)**

Input: A number ***x***

Output: "**Yes**" if ***x*** is a natural number, else "**No**"

Algorithm:

```
if $x < 0$,  then return "No"
else
   if $x = 0$,  then return "Yes"
   else return Natural( $x - 1$ )
```

Example 5: Algorithm for testing whether or not an expression ***w*** is a proposition(propositional form)

Algorithm 5 Proposition(a string ***w*** **)**

Input: A string ***w***

Output: "**Yes**" if ***w*** is a proposition, else "**No**"

Algorithm:

```
if $w$ is $1$(true), $0$(false), or a propositional variable, then return "Yes"
else if $w = \sim w_1$, then return Proposition($w_1$)
  else
    if ( $w = w_1 \vee w_2$ or $w_1 \wedge w_2$ or $w_1 \rightarrow w_2$ or $w_1 \leftrightarrow w_2$ ) and
      Proposition($w_1$) = Yes and  Proposition($w_2$) = Yes
    then return Yes
    else return No
end
```

G.KAMALAKANNAN MCA., M.Tech., (PhD)
HQMHSS, Chromepet, Chennai.
8939665453