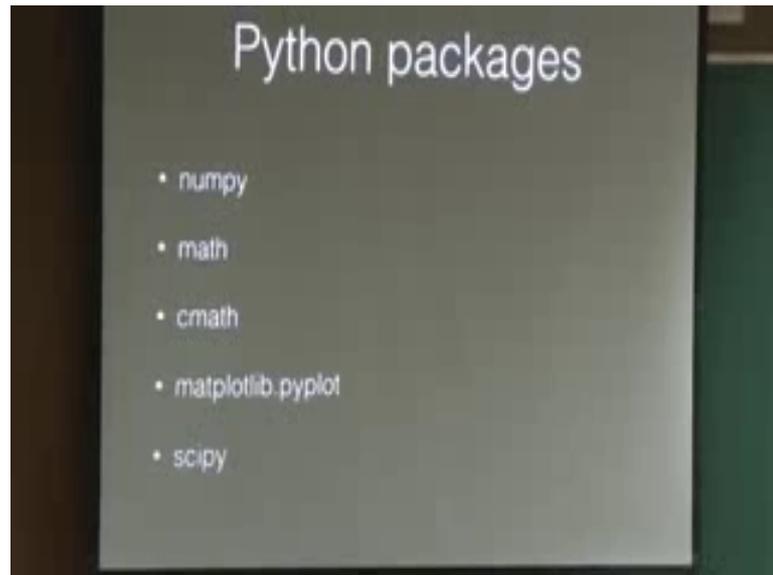


Computational Science and Engineering using Python
Prof. Mahendra K. Verma
Department of Physics
Indian Institute of Technology, Kanpur

Lecture – 5A
Python Packages; Programming

(Refer Slide Time: 00:13)



Now, I also want to tell about Python packages. I have been delaying it for a while because, I thought you should take one id at a time, but now it is a good time to discuss about these packages. In a programming language if it a rich programming language it has many features. And when you start working on it, you do not include all the features. Why should not you include all the features, because it will occupy your ram know all the function everything will occupy your memory. It will make it slow. If I put all every Python packages inside it will fill up your laptop.

So, it is a good idea to use, once which you need. The packages which we need for this course are numpy, which you already know numpy numerical Python. So, this is for arrays and numerical functions like sin, square root, all that factorial; factorial it does not do I just checked numpy does not affect factorial. Numpy will work with arrays, its strength is arrays, and I discussed that in the class.

So, if you want to use arrays and numpy and operations on arrays. If you want to just use math on single variables like factorial 5, then you have to use math. So, lot of their lot of

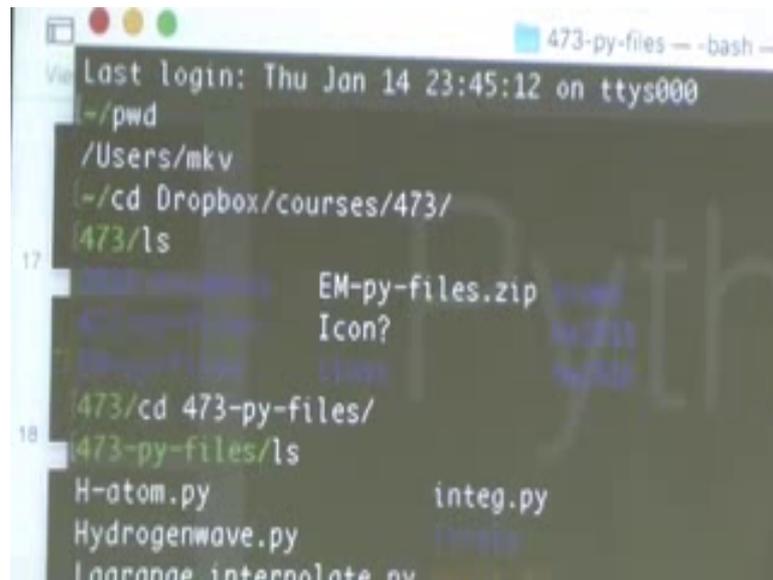
common functions I mean `sin` of a array is same as `sin` of I mean I can make a array of one number. So, `sin` is accepted `sin` of 5 will be accepted, but not factorial of 5. `C math` is for complex numbers. `Math` has quite a bit of it `numpy` also sometimes you may just not need `c math`, but if you see a complex numbers are not working then you import `c math`.

I will tell you how to import it. There is something called `matplotlib`. So, this is for plotting, which we will do in the next class. Python can do beautiful plots. And you can plot 3D contour plots lot of it. So, this is really nice. So, I will tell you this one, and `SciPy`. So, this all scientific Python and eigen values, is part of scientific Python, is a part of linear algebra package of scientific Python. Differential equation solvers are part of `SciPy`. So, lot of scientific Python their packages, and we will use some of them; matrix stuff, woody solvers we will do that one.

Again, I must say that I will not hand hold you. You have to learn few things yourselves, but just to kind of get you on because, yesterdays lab people were not even knowing how to save in a file. All that is you must learn it as UNIX stuffs you should learn it. In fact, I strongly suggest that install UNIX in your machine. No, excuse that it works in windows it does not work here; it is not my problem, you told me this something which you have to handle it.

So, everybody knows what is `p w d` present working directory. So, it is tells with the present what is what is your current directory, but I have to go to directory where I have my code. So, I can just start Python from n terminal, but if you are if a program is written in a file then you have to go that particular directory then only you can run it; otherwise you do not know where to search.

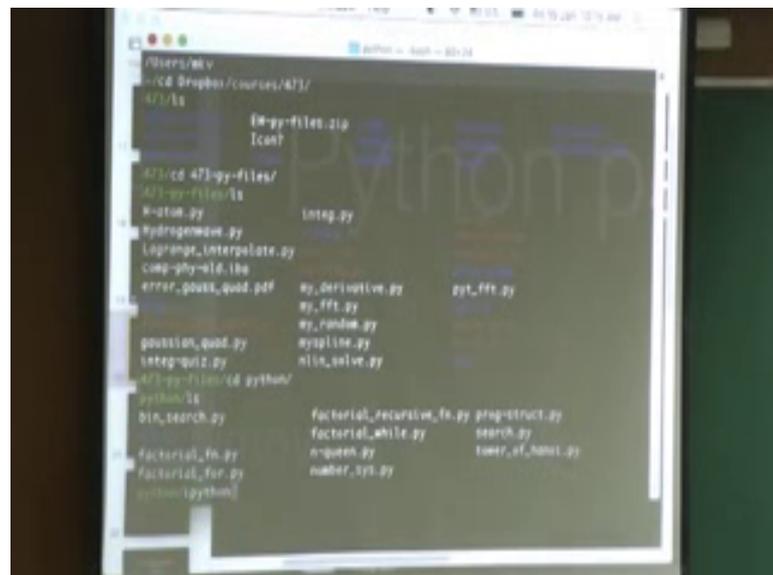
(Refer Slide Time: 03:49)



```
473-py-files -- bash --
Last login: Thu Jan 14 23:45:12 on ttys000
~/pwd
/Users/mkv
~/cd Dropbox/courses/473/
473/ls
EM-py-files.zip
Icon?
473/cd 473-py-files/
473-py-files/ls
H-atom.py          integ.py
Hydrogenwave.py
Lagrange_interpolate.py
```

But let us start doing Python here. Let us go to that directory; because I need it for something else is there. One trick is if you put a tab, if you not batch profile if it is not scrued up, then it will just take the complete the string 473. So, I can always see what is there I is not easy to see from here.

(Refer Slide Time: 04:28)



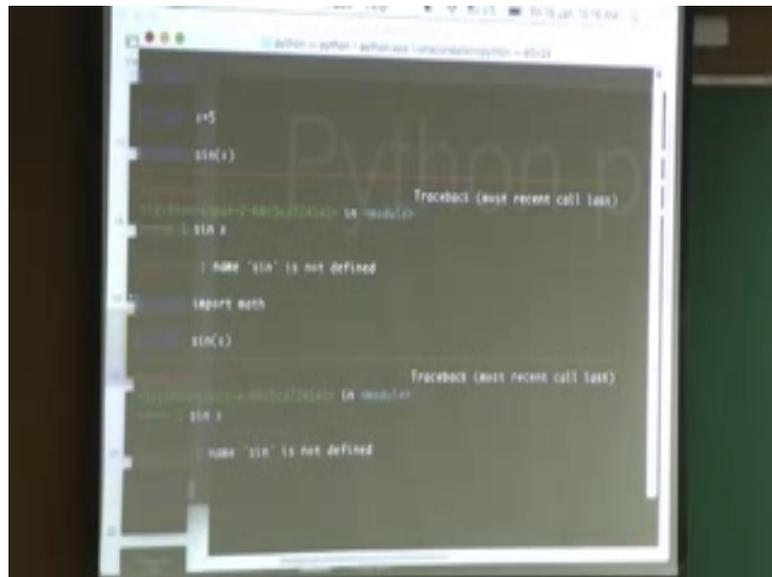
```
/Users/mkv
~/cd Dropbox/courses/473/
473/ls
EM-py-files.zip
Icon?
473/cd 473-py-files/
473-py-files/ls
H-atom.py          integ.py
Hydrogenwave.py
Lagrange_interpolate.py
comp-ply-nls.lhs
error_gauss_quad.pdf  py_derivative.py  zyt_fft.py
                        py_fft.py
                        py_random.py
                        spline.py
                        solve_nls.py
gaussian_quad.py
integrate2.py
473-py-files/cd python/
python ls
bin_search.py          factorial_recursive_fn.py prog_struct.py
                        factorial_while.py  search.py
                        n_queen.py
                        number_tst.py
factorial_fn.py
factorial_for.py
python (python)
```

So, l s will tell you where it is. What are the files inside? And this is what were I want to be because my codes are here. I just start I Python. Please recall that I had put minus minus PyLab before. So, PyLab imports numpy as well as math stuff. So, it is useful you

also part of some of well it does not import everything, but very useful thing it imports, including eigen. Eigen values right it worked, but if you just do not do it then, your this will what will come. Now you cannot see this blue color, but let us just start.

So, it is user stuff x is equal to 5, but I want to say \sin of x . Do you think it will give the answer? \sin not defined, because it does not know what is \sin ? So, I have to import numpy or you could also import math math will work. So, you can import both. So, let us import math and see whether it will work. If I say import math it will import everything of math. Now what should I say, I imported math $\sin x$ will it work. So, it will not work. So, it is a part of math package.

(Refer Slide Time: 05:47)



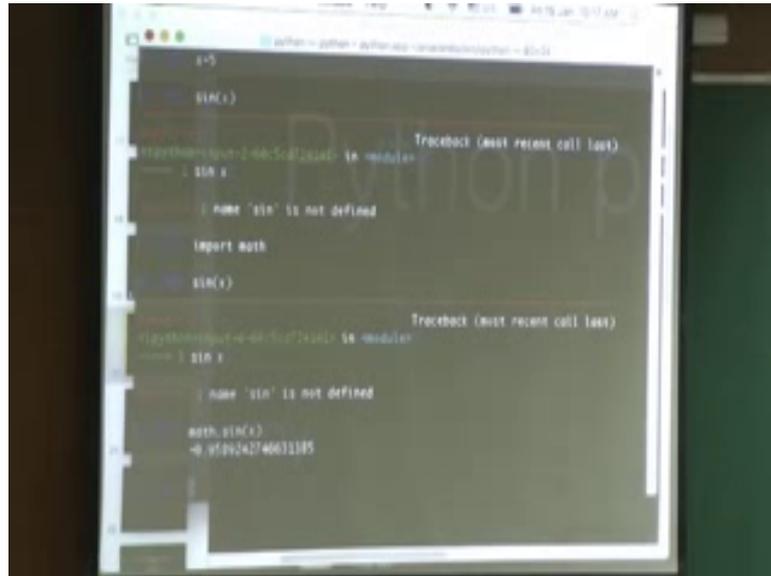
```
Python 3.7.4 Shell
>>> x=5
>>> sin(x)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'sin' is not defined

>>> import math
>>> sin(x)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'sin' is not defined

>>> math.sin(x)
0.9612616959386416
```

So, I have to say `math dot sin`, it is part of it. So, you have to give the Kanpur, IIT Kanpur then home. So, you have to go in that sequence. So, you have to say `math dot sin` of x this works ok.

(Refer Slide Time: 06:05)



```
Python 3.7.4 Shell
>>> x=5
>>> sin(x)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
    sin(x)
NameError: name 'sin' is not defined

>>> import math
>>> sin(x)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
    sin(x)
NameError: name 'sin' is not defined

>>> math.sin(x)
-0.959242746611185
```

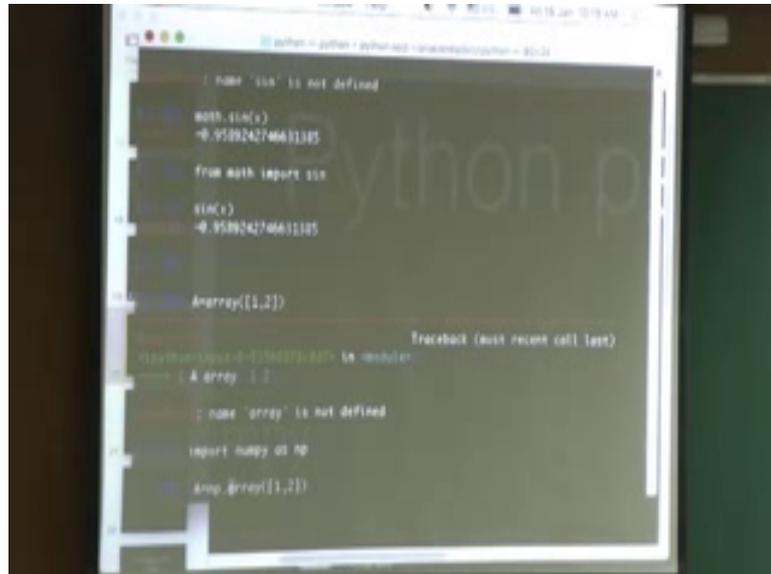
So, we import a package then you have to from package you have to save this. There is another way to do it. So, I do not want to write this math dot. So, there is a way to do it. So, this you should remember I mean, these are simple English, but you should think logic. So, let us I want to import cos, but right now math everything is imported, but I can just I could if I had done this, from math import sin.

So, it will import sin function. In fact, it imports it, it compiles it, it has a object file sitting there if you know this object stuff, so it is sitting there, if I do this. Now I can use sin of x. So, if I say sin of x it will work. So, sometimes I may use this one or you may use this one or sometimes you may use this one. So, if you import that is, what you do not want to import everything then you should say from import, from math import whatever we need know necessary to import everything where that will occupy your memory.

Here sometimes is not a good idea to possess lot of things we just possess what you want Buddas type, but what if I say array will it work; no a equal to array, I put work or no work array does not know. So, what should I do I import numpy, but it is good idea to import numpy. So, numpy is a long word. Sometimes, the names of the functions are very packages are very long. So, I want to make a short encomium. So, it is usual to import as. So, you do not have to say numpy dot. So, we use np shorthand, so, np now if I say. So, I could have said now it. So, numpy name is np now it has been renamed to np.

So, I should just put np dot, and it will work. So, instead of numpy it is you know it takes time to write. So, np will work.

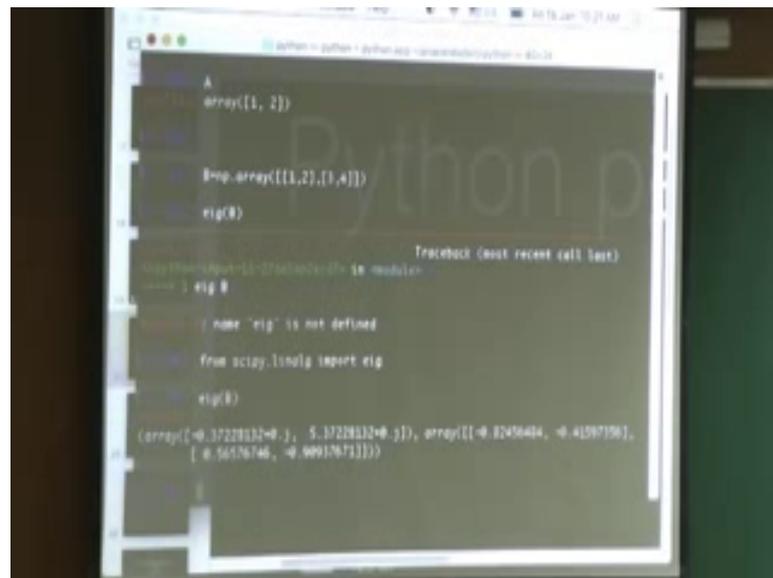
(Refer Slide Time: 08:11)



```
Python Shell
> name 'sin' is not defined
>>> math.sin(x)
-0.958924274661185
>>> from math import sin
>>> sin(x)
-0.958924274661185
>>>
>>> Array([1,2])
Traceback (most recent call last)
Python Shell In [2]:
----> 1: Array([1,2])
      2:
      3:
      4:
      5:
      6:
      7:
      8:
      9:
     10:
     11:
     12:
     13:
     14:
     15:
     16:
     17:
     18:
     19:
     20:
     21:
     22:
     23:
     24:
     25:
     26:
     27:
     28:
     29:
     30:
     31:
     32:
     33:
     34:
     35:
     36:
     37:
     38:
     39:
     40:
     41:
     42:
     43:
     44:
     45:
     46:
     47:
     48:
     49:
     50:
     51:
     52:
     53:
     54:
     55:
     56:
     57:
     58:
     59:
     60:
     61:
     62:
     63:
     64:
     65:
     66:
     67:
     68:
     69:
     70:
     71:
     72:
     73:
     74:
     75:
     76:
     77:
     78:
     79:
     80:
     81:
     82:
     83:
     84:
     85:
     86:
     87:
     88:
     89:
     90:
     91:
     92:
     93:
     94:
     95:
     96:
     97:
     98:
     99:
    100:
    101:
    102:
    103:
    104:
    105:
    106:
    107:
    108:
    109:
    110:
    111:
    112:
    113:
    114:
    115:
    116:
    117:
    118:
    119:
    120:
    121:
    122:
    123:
    124:
    125:
    126:
    127:
    128:
    129:
    130:
    131:
    132:
    133:
    134:
    135:
    136:
    137:
    138:
    139:
    140:
    141:
    142:
    143:
    144:
    145:
    146:
    147:
    148:
    149:
    150:
    151:
    152:
    153:
    154:
    155:
    156:
    157:
    158:
    159:
    160:
    161:
    162:
    163:
    164:
    165:
    166:
    167:
    168:
    169:
    170:
    171:
    172:
    173:
    174:
    175:
    176:
    177:
    178:
    179:
    180:
    181:
    182:
    183:
    184:
    185:
    186:
    187:
    188:
    189:
    190:
    191:
    192:
    193:
    194:
    195:
    196:
    197:
    198:
    199:
    200:
    201:
    202:
    203:
    204:
    205:
    206:
    207:
    208:
    209:
    210:
    211:
    212:
    213:
    214:
    215:
    216:
    217:
    218:
    219:
    220:
    221:
    222:
    223:
    224:
    225:
    226:
    227:
    228:
    229:
    230:
    231:
    232:
    233:
    234:
    235:
    236:
    237:
    238:
    239:
    240:
    241:
    242:
    243:
    244:
    245:
    246:
    247:
    248:
    249:
    250:
    251:
    252:
    253:
    254:
    255:
    256:
    257:
    258:
    259:
    260:
    261:
    262:
    263:
    264:
    265:
    266:
    267:
    268:
    269:
    270:
    271:
    272:
    273:
    274:
    275:
    276:
    277:
    278:
    279:
    280:
    281:
    282:
    283:
    284:
    285:
    286:
    287:
    288:
    289:
    290:
    291:
    292:
    293:
    294:
    295:
    296:
    297:
    298:
    299:
    300:
    301:
    302:
    303:
    304:
    305:
    306:
    307:
    308:
    309:
    310:
    311:
    312:
    313:
    314:
    315:
    316:
    317:
    318:
    319:
    320:
    321:
    322:
    323:
    324:
    325:
    326:
    327:
    328:
    329:
    330:
    331:
    332:
    333:
    334:
    335:
    336:
    337:
    338:
    339:
    340:
    341:
    342:
    343:
    344:
    345:
    346:
    347:
    348:
    349:
    350:
    351:
    352:
    353:
    354:
    355:
    356:
    357:
    358:
    359:
    360:
    361:
    362:
    363:
    364:
    365:
    366:
    367:
    368:
    369:
    370:
    371:
    372:
    373:
    374:
    375:
    376:
    377:
    378:
    379:
    380:
    381:
    382:
    383:
    384:
    385:
    386:
    387:
    388:
    389:
    390:
    391:
    392:
    393:
    394:
    395:
    396:
    397:
    398:
    399:
    400:
    401:
    402:
    403:
    404:
    405:
    406:
    407:
    408:
    409:
    410:
    411:
    412:
    413:
    414:
    415:
    416:
    417:
    418:
    419:
    420:
    421:
    422:
    423:
    424:
    425:
    426:
    427:
    428:
    429:
    430:
    431:
    432:
    433:
    434:
    435:
    436:
    437:
    438:
    439:
    440:
    441:
    442:
    443:
    444:
    445:
    446:
    447:
    448:
    449:
    450:
    451:
    452:
    453:
    454:
    455:
    456:
    457:
    458:
    459:
    460:
    461:
    462:
    463:
    464:
    465:
    466:
    467:
    468:
    469:
    470:
    471:
    472:
    473:
    474:
    475:
    476:
    477:
    478:
    479:
    480:
    481:
    482:
    483:
    484:
    485:
    486:
    487:
    488:
    489:
    490:
    491:
    492:
    493:
    494:
    495:
    496:
    497:
    498:
    499:
    500:
    501:
    502:
    503:
    504:
    505:
    506:
    507:
    508:
    509:
    510:
    511:
    512:
    513:
    514:
    515:
    516:
    517:
    518:
    519:
    520:
    521:
    522:
    523:
    524:
    525:
    526:
    527:
    528:
    529:
    530:
    531:
    532:
    533:
    534:
    535:
    536:
    537:
    538:
    539:
    540:
    541:
    542:
    543:
    544:
    545:
    546:
    547:
    548:
    549:
    550:
    551:
    552:
    553:
    554:
    555:
    556:
    557:
    558:
    559:
    560:
    561:
    562:
    563:
    564:
    565:
    566:
    567:
    568:
    569:
    570:
    571:
    572:
    573:
    574:
    575:
    576:
    577:
    578:
    579:
    580:
    581:
    582:
    583:
    584:
    585:
    586:
    587:
    588:
    589:
    590:
    591:
    592:
    593:
    594:
    595:
    596:
    597:
    598:
    599:
    600:
    601:
    602:
    603:
    604:
    605:
    606:
    607:
    608:
    609:
    610:
    611:
    612:
    613:
    614:
    615:
    616:
    617:
    618:
    619:
    620:
    621:
    622:
    623:
    624:
    625:
    626:
    627:
    628:
    629:
    630:
    631:
    632:
    633:
    634:
    635:
    636:
    637:
    638:
    639:
    640:
    641:
    642:
    643:
    644:
    645:
    646:
    647:
    648:
    649:
    650:
    651:
    652:
    653:
    654:
    655:
    656:
    657:
    658:
    659:
    660:
    661:
    662:
    663:
    664:
    665:
    666:
    667:
    668:
    669:
    670:
    671:
    672:
    673:
    674:
    675:
    676:
    677:
    678:
    679:
    680:
    681:
    682:
    683:
    684:
    685:
    686:
    687:
    688:
    689:
    690:
    691:
    692:
    693:
    694:
    695:
    696:
    697:
    698:
    699:
    700:
    701:
    702:
    703:
    704:
    705:
    706:
    707:
    708:
    709:
    710:
    711:
    712:
    713:
    714:
    715:
    716:
    717:
    718:
    719:
    720:
    721:
    722:
    723:
    724:
    725:
    726:
    727:
    728:
    729:
    730:
    731:
    732:
    733:
    734:
    735:
    736:
    737:
    738:
    739:
    740:
    741:
    742:
    743:
    744:
    745:
    746:
    747:
    748:
    749:
    750:
    751:
    752:
    753:
    754:
    755:
    756:
    757:
    758:
    759:
    760:
    761:
    762:
    763:
    764:
    765:
    766:
    767:
    768:
    769:
    770:
    771:
    772:
    773:
    774:
    775:
    776:
    777:
    778:
    779:
    780:
    781:
    782:
    783:
    784:
    785:
    786:
    787:
    788:
    789:
    790:
    791:
    792:
    793:
    794:
    795:
    796:
    797:
    798:
    799:
    800:
    801:
    802:
    803:
    804:
    805:
    806:
    807:
    808:
    809:
    810:
    811:
    812:
    813:
    814:
    815:
    816:
    817:
    818:
    819:
    820:
    821:
    822:
    823:
    824:
    825:
    826:
    827:
    828:
    829:
    830:
    831:
    832:
    833:
    834:
    835:
    836:
    837:
    838:
    839:
    840:
    841:
    842:
    843:
    844:
    845:
    846:
    847:
    848:
    849:
    850:
    851:
    852:
    853:
    854:
    855:
    856:
    857:
    858:
    859:
    860:
    861:
    862:
    863:
    864:
    865:
    866:
    867:
    868:
    869:
    870:
    871:
    872:
    873:
    874:
    875:
    876:
    877:
    878:
    879:
    880:
    881:
    882:
    883:
    884:
    885:
    886:
    887:
    888:
    889:
    890:
    891:
    892:
    893:
    894:
    895:
    896:
    897:
    898:
    899:
    900:
    901:
    902:
    903:
    904:
    905:
    906:
    907:
    908:
    909:
    910:
    911:
    912:
    913:
    914:
    915:
    916:
    917:
    918:
    919:
    920:
    921:
    922:
    923:
    924:
    925:
    926:
    927:
    928:
    929:
    930:
    931:
    932:
    933:
    934:
    935:
    936:
    937:
    938:
    939:
    940:
    941:
    942:
    943:
    944:
    945:
    946:
    947:
    948:
    949:
    950:
    951:
    952:
    953:
    954:
    955:
    956:
    957:
    958:
    959:
    960:
    961:
    962:
    963:
    964:
    965:
    966:
    967:
    968:
    969:
    970:
    971:
    972:
    973:
    974:
    975:
    976:
    977:
    978:
    979:
    980:
    981:
    982:
    983:
    984:
    985:
    986:
    987:
    988:
    989:
    990:
    991:
    992:
    993:
    994:
    995:
    996:
    997:
    998:
    999:
    1000:
    1001:
    1002:
    1003:
    1004:
    1005:
    1006:
    1007:
    1008:
    1009:
    1010:
    1011:
    1012:
    1013:
    1014:
    1015:
    1016:
    1017:
    1018:
    1019:
    1020:
    1021:
    1022:
    1023:
    1024:
    1025:
    1026:
    1027:
    1028:
    1029:
    1030:
    1031:
    1032:
    1033:
    1034:
    1035:
    1036:
    1037:
    1038:
    1039:
    1040:
    1041:
    1042:
    1043:
    1044:
    1045:
    1046:
    1047:
    1048:
    1049:
    1050:
    1051:
    1052:
    1053:
    1054:
    1055:
    1056:
    1057:
    1058:
    1059:
    1060:
    1061:
    1062:
    1063:
    1064:
    1065:
    1066:
    1067:
    1068:
    1069:
    1070:
    1071:
    1072:
    1073:
    1074:
    1075:
    1076:
    1077:
    1078:
    1079:
    1080:
    1081:
    1082:
    1083:
    1084:
    1085:
    1086:
    1087:
    1088:
    1089:
    1090:
    1091:
    1092:
    1093:
    1094:
    1095:
    1096:
    1097:
    1098:
    1099:
    1100:
    1101:
    1102:
    1103:
    1104:
    1105:
    1106:
    1107:
    1108:
    1109:
    1110:
    1111:
    1112:
    1113:
    1114:
    1115:
    1116:
    1117:
    1118:
    1119:
    1120:
    1121:
    1122:
    1123:
    1124:
    1125:
    1126:
    1127:
    1128:
    1129:
    1130:
    1131:
    1132:
    1133:
    1134:
    1135:
    1136:
    1137:
    1138:
    1139:
    1140:
    1141:
    1142:
    1143:
    1144:
    1145:
    1146:
    1147:
    1148:
    1149:
    1150:
    1151:
    1152:
    1153:
    1154:
    1155:
    1156:
    1157:
    1158:
    1159:
    1160:
    1161:
    1162:
    1163:
    1164:
    1165:
    1166:
    1167:
    1168:
    1169:
    1170:
    1171:
    1172:
    1173:
    1174:
    1175:
    1176:
    1177:
    1178:
    1179:
    1180:
    1181:
    1182:
    1183:
    1184:
    1185:
    1186:
    1187:
    1188:
    1189:
    1190:
    1191:
    1192:
    1193:
    1194:
    1195:
    1196:
    1197:
    1198:
    1199:
    1200:
    1201:
    1202:
    1203:
    1204:
    1205:
    1206:
    1207:
    1208:
    1209:
    1210:
    1211:
    1212:
    1213:
    1214:
    1215:
    1216:
    1217:
    1218:
    1219:
    1220:
    1221:
    1222:
    1223:
    1224:
    1225:
    1226:
    1227:
    1228:
    1229:
    1230:
    1231:
    1232:
    1233:
    1234:
    1235:
    1236:
    1237:
    1238:
    1239:
    1240:
    1241:
    1242:
    1243:
    1244:
    1245:
    1246:
    1247:
    1248:
    1249:
    1250:
    1251:
    1252:
    1253:
    1254:
    1255:
    1256:
    1257:
    1258:
    1259:
    1260:
    1261:
    1262:
    1263:
    1264:
    1265:
    1266:
    1267:
    1268:
    1269:
    1270:
    1271:
    1272:
    1273:
    1274:
    1275:
    1276:
    1277:
    1278:
    1279:
    1280:
    1281:
    1282:
    1283:
    1284:
    1285:
    1286:
    1287:
    1288:
    1289:
    1290:
    1291:
    1292:
    1293:
    1294:
    1295:
    1296:
    1297:
    1298:
    1299:
    1300:
    1301:
    1302:
    1303:
    1304:
    1305:
    1306:
    1307:
    1308:
    1309:
    1310:
    1311:
    1312:
    1313:
    1314:
    1315:
    1316:
    1317:
    1318:
    1319:
    1320:
    1321:
    1322:
    1323:
    1324:
    1325:
    1326:
    1327:
    1328:
    1329:
    1330:
    1331:
    1332:
    1333:
    1334:
    1335:
    1336:
    1337:
    1338:
    1339:
    1340:
    1341:
    1342:
    1343:
    1344:
    1345:
    1346:
    1347:
    1348:
    1349:
    1350:
    1351:
    1352:
    1353:
    1354:
    1355:
    1356:
    1357:
    1358:
    1359:
    1360:
    1361:
    1362:
    1363:
    1364:
    1365:
    1366:
    1367:
    1368:
    1369:
    1370:
    1371:
    1372:
    1373:
    1374:
    1375:
    1376:
    1377:
    1378:
    1379:
    1380:
    1381:
    1382:
    1383:
    1384:
    1385:
    1386:
    1387:
    1388:
    1389:
    1390:
    1391:
    1392:
    1393:
    1394:
    1395:
    1396:
    1397:
    1398:
    1399:
    1400:
    1401:
    1402:
    1403:
    1404:
    1405:
    1406:
    1407:
    1408:
    1409:
    1410:
    1411:
    1412:
    1413:
    1414:
    1415:
    1416:
    1417:
    1418:
    1419:
    1420:
    1421:
    1422:
    1423:
    1424:
    1425:
    1426:
    1427:
    1428:
    1429:
    1430:
    1431:
    1432:
    1433:
    1434:
    1435:
    1436:
    1437:
    1438:
    1439:
    1440:
    1441:
    1442:
    1443:
    1444:
    1445:
    1446:
    1447:
    1448:
    1449:
    1450:
    1451:
    1452:
    1453:
    1454:
    1455:
    1456:
    1457:
    1458:
    1459:
    1460:
    1461:
    1462:
    1463:
    1464:
    1465:
    1466:
    1467:
    1468:
    1469:
    1470:
    1471:
    1472:
    1473:
    1474:
    1475:
    1476:
    1477:
    1478:
    1479:
    1480:
    1481:
    1482:
    1483:
    1484:
    1485:
    1486:
    1487:
    1488:
    1489:
    1490:
    1491:
    1492:
    1493:
    1494:
    1495:
    1496:
    1497:
    1498:
    1499:
    1500:
    1501:
    1502:
    1503:
    1504:
    1505:
    1506:
    1507:
    1508:
    1509:
    1510:
    1511:
    1512:
    1513:
    1514:
    1515:
    1516:
    1517:
    1518:
    1519:
    1520:
    1521:
    1522:
    1523:
    1524:
    1525:
    1526:
    1527:
    1528:
    1529:
    1530:
    1531:
    1532:
    1533:
    1534:
    1535:
    1536:
    1537:
    1538:
    1539:
    1540:
    1541:
    1542:
    1543:
    1544:
    1545:
    1546:
    1547:
    1548:
    1549:
    1550:
    1551:
    1552:
    1553:
    1554:
    1555:
    1556:
    1557:
    1558:
    1559:
    1560:
    1561:
    1562:
    1563:
    1564:
    1565:
    1566:
    1567:
    1568:
    1569:
    1570:
    1571:
    1572:
    1573:
    1574:
    1575:
    1576:
    1577:
    1578:
    1579:
    1580:
    1581:
    1582:
    1583:
    1584:
    1585:
    1586:
    1587:
    1588:
    1589:
    1590:
    1591:
    1592:
    1593:
    1594:
    1595:
    1596:
    1597:
    1598:
    1599:
    1600:
    1601:
    1602:
    1603:
    1604:
    1605:
    1606:
    1607:
    1608:
    1609:
    1610:
    1611:
    1612:
    1613:
    1614:
    1615:
    1616:
    1617:
    1618:
    1619:
    1620:
    1621:
    1622:
    1623:
    1624:
    1625:
    1626:
    1627:
    1628:
    1629:
    1630:
    1631:
    1632:
    1633:
    1634:
    1635:
    1636:
    1637:
    1638:
    1639:
    1640:
    1641:
    1642:
    1643:
    1644:
    1645:
    1646:
    1647:
```

So, SciPy has many many libraries; a linear algebra of that. So, I say from SciPy dot linear algebra. So, it is again within you are going down import eig. So, you can import in many ways. So, from this package I import eig. Now if you say eig b this is how you import in different ways. If you want to import only few things you import from this import you want to import everything you can also import the package name. So, it comes in many flavors.

(Refer Slide Time: 09:28)



```
A
array([1, 2])

Inp.array([[1,2],[3,4]])
eig(B)

Traceback (most recent call last):
  File "Python p", line 1, in <module>
    eig(B)
NameError: name 'eig' is not defined

from scipy.linalg import eig
eig(B)

(array([-0.37228132+0.j,  5.37228132+0.j]), array([[-0.02456484, -0.41597196],
        [ 0.56576746, -0.99917671]]))
```

So, I listed some flavors here. So, is things are clear. Import is a way to do it, but there are many ways to import it and I just think about it, and I will list it here.

(Refer Slide Time: 09:58)

```
import math
from math import factorial

import numpy
import numpy as np

from scipy import linalg
from scipy.linalg import eig
from matplotlib import pyplot as plt
```

In this you will find in this sorts, like import math or from math import factorial. So, it will import only this: import numpy or import numpy as something this is shorthand from SciPy import linear algebra. So, you could I could import all of linear algebra, dot product I do not remember right now, but you can do lot of stuff with that. And linear algebra also has inversion of a matrix, lot of stuff. So, SciPy So, I could also said I only want eig. So, SciPy dot lin, so I could do this.

Now we will move to plotting. So, this is what you should put always you know from plot matplotlib import pyplot. So, matplotlib is a very big library. So, from that I need only some of it, which is looks like MATLAB. So, for us it is a (Refer Time: 10:59) to use pyplot and I am going to use instead of writing the 6 letter word I will use plt. So, this will be my standard plt dot. It turns out you can plot right now in PyLab. So, PyLab is very useful for doing most of the operations, on the screen. So, thats why I did not do import anything it was all imported, I could happily work so, far.

But, if you want to use in a file, when you import in a file, there could be there was a problem yesterday. So, array was defined inside a file. So, you heard to say array inside a file. So, when you say run it is running that file. So, within a file you have to import everything whatever is required within that file. So, that is, why you need this stuff; inside a file you need even though PyLab eig minus means PyLab within a file you need to import whatever is required. So, just to give you an example.

(Refer Slide Time: 12:11)

A photograph of a computer monitor displaying a code editor window with Python code. The code includes imports for numpy and matplotlib, and a plot function. The code is as follows: 'import numpy as np', 'from matplotlib.figure import Figure', 'from matplotlib.backends.backend_agg import FigureCanvasAgg as FigureCanvas', 'def plot(x, y):', ' fig = Figure()', ' canvas = FigureCanvas(fig)', ' ax = fig.gca()', ' ax.plot(x, y)', ' canvas.draw()', ' return fig', 'if __name__ == '__main__':', ' x = np.linspace(0, 10, 100)', ' y = np.sin(x)', ' plot(x, y)'

So, at the top normally it is a good idea to, I mean I need a numpy. In fact, this was not required here I think I believe. So, matplotlib these are I you put in top. And so, things will work after that. So, I have used np dot.

So, you know the engineering, I mean or you know the tools you have the tools, but now how to write good codes, or how to write workable code first good will come after a while. So, there are tips. In fact, there are books written on it; huge number of books in internet you say how to write good programs you will get visibly good sites, and I think you should read.

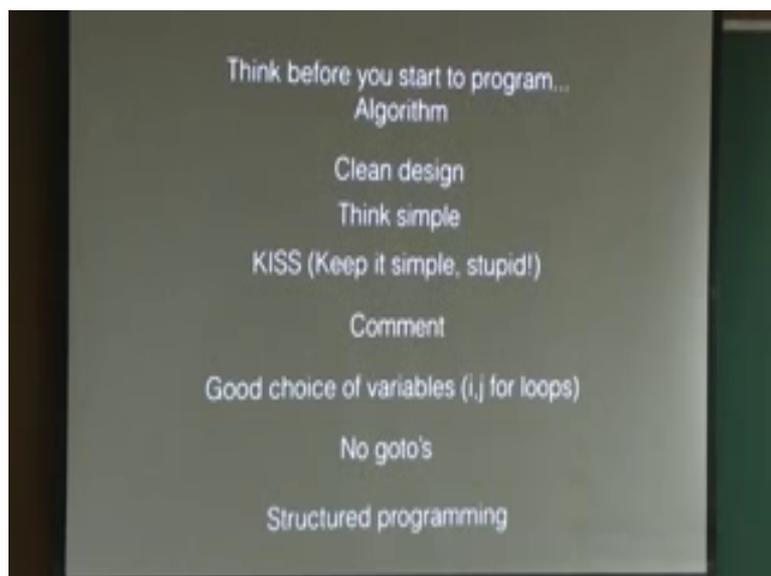
So, I will give you some tips; which were I will tell you in 5 minutes or 10 minutes.

(Refer Slide Time: 12:55)



But, you should pay attention to it is critical, just it is not like I can scribble something computer will take it. Computer is a dumb machine, but very powerful machine. So, it can also eat you up so, but you have to write properly to make it work. If you do not write properly, it will not work; I mean you cannot be frustrated it is the syntax error, I mean you have to fix the syntax error. You should work in such a way that there is no syntax error. So, the tips one important point I did not see anybody while few people were writing on a piece of paper.

(Refer Slide Time: 13:38)



But, please take a note-book in the lab session; do not go empty handed. And you must scribble to get the idea, in fact, I could not I should not I want to write a code without thinking like this in a on the piece of paper. You must always think before you start coding. For a big code which I wrote I thought for 1 year, this is what we think a lot how to design a code. The good big programs 6 months no coding, you just think of design first. So, do not need to I mean you will be doing mistakes if you start coding. So, like you do not start building a, big complex order of brick, order sand you do not do it you know you first design the building.

So, this is critical you should even for simple things like searching in the last class, you should think before you start coding. It become should become a habit a clean design you do not want to say well, now I need it so, I will add a room; it will look ugly in a building. Simply in a code you should not simply just create a variable indentation is not there; Python is very good that way it will enforce indentation. I seen students writing everything from a column of 0 footer and codes never never never do it.

Indentation is good for you, forget about others. You cannot understand the code if you just start from everything from column 0. The code should be understandable. so clean design is critical; think simple; do not think complicated; you cannot write good code with complex thinking, if there are also who have good for physics also, I do believe that we think too complicated and write too many junk papers; think simple.

So, there is a phrase in computer science it is called kiss, it means keep it simple stupid. So, keep it simple. Some clever programmers want to put in some clever things in the code, it is not advised because you yourself will come and look at it what did I do. So, even though it may take one more line keep it simple.

Comments: you should comment your code. What is input? What is output? Few lines you should comment. I do not know last time what variables you were using. So, it should be good choice of variables for example, I will do one illustration, I will try to write a code for searching. So, my variables must be chosen with some thought like factorial; I should not say x equal to something. So, factorial I will put fact I did not want to use the word factorial for my variable y, it will conflict with the name of the function and that is even though a Python may work I do not know I did not test it, but is not a good choice. So, this is important at least two students asked me goto's

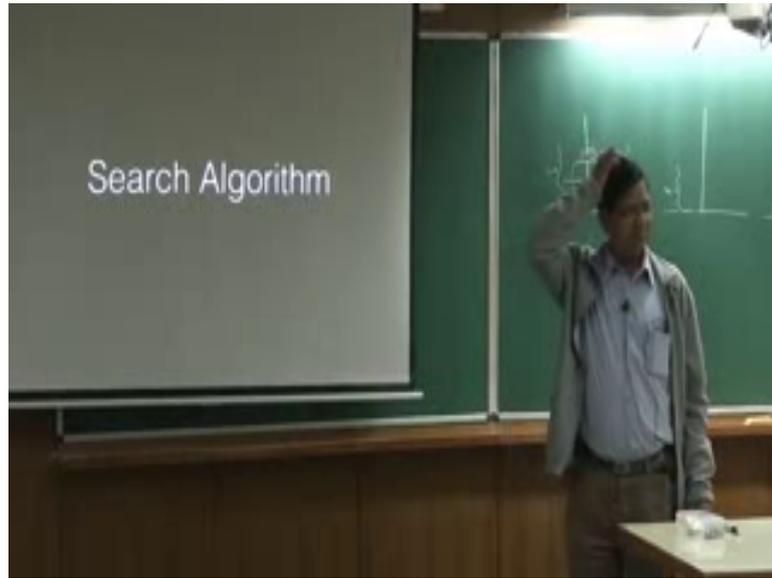
Goto is banned. Why is goto banned? Goto is banned because, suppose I say after the class you go to Kanpur University that is, what you will do in the next class. That is means goto, goto will take you somewhere very far away without logic and from Kanpur University you will goto mirut for the third class. So, this is not a good design right, goto is you should go in the neighborhood. Next line is what you should goto unless it is told. So, if you need to go somewhere uses function. So, function will do the job.

So, I say on the whole I take a vacation to that place and I will finish it off and comeback. Goto's are not there in I mean Python you I do not think it is there, there is no line numbering so, it is not possible; well I believe so I mean, there may be trick to do it, but you do not need it; you should not needing it is banned in any language even if it is given to you should not use it.

So, the way to code is called structured programming. So, your code should look like a structure: like a building complex. You should you should look clean you should look nice beautiful and that is how it should be and there are ways to the more sophisticated ways to do it you do not need for this course this is called object oriented programming, that makes it clean and you can write large codes, I mean if you want to write one like line code you need more thoughts and more tools. So, object oriented programming is achieves that, but we do not need for this course. Any question on this?

So, What I want to do it quickly? Just illustrate how to write a code. We know we have all the tools now to write simple codes.

(Refer Slide Time: 18:24)

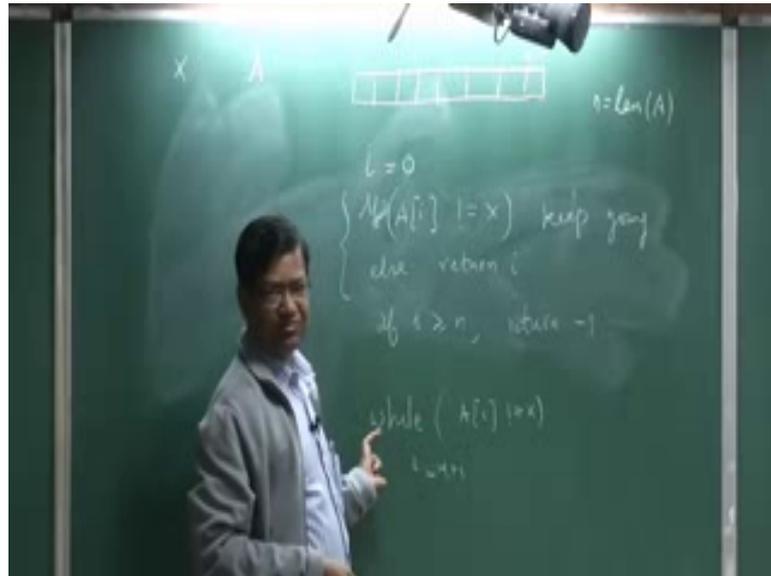


So, we will do search algorithm. Now this does not look like a physics part; true it is not really a physics problem, but it is a very important computer science problem, because you use it for all the time, searching inside a like Google is searching; most of it is searching, but in a directory I am searching. So, search is very useful algorithm. So, let us try to write program which will search in an array.

So, I limit my objective, I say I am going to look for program, or I am going to write a program which will search a variable x . So, I should say what variable must search x , I am going to rename that variable. X is a good choice of variable, which is their inside an array a or not. I will write a function, function is reusable, so that is my objectives. So, if it is it is a part of the array a , it should give me the index where it is. If it is not a part of the array a , then I said give minus 1; unfortunately Python minus 1 means an index. The last index, but let us just work with that, with that understanding that minus 1 means not in the array. So, how should I do it?

First, do not start coding. Think of what you should do. So, it is a good idea to make pictures.

(Refer Slide Time: 20:01)



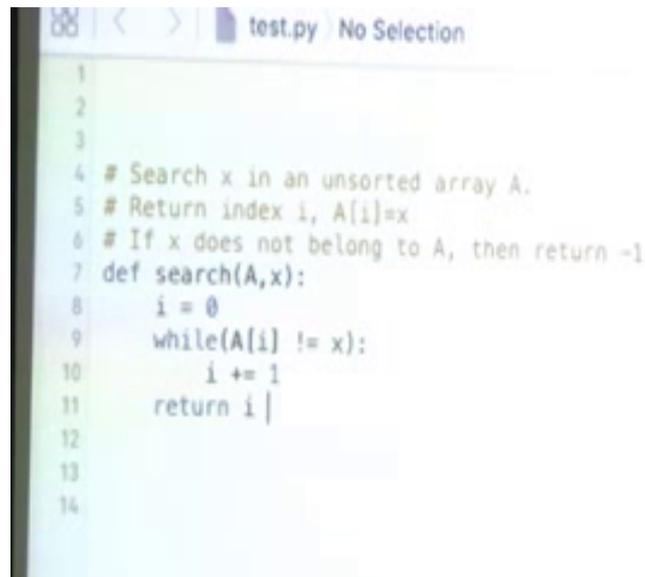
So, this is my array a ; like this. So, what should I do, is unsorted array. Some array is given to you while I do not know sorted or unsorted. So, is simple; I start looking from first place, if it is x I return the index, if not then go to a second, third, fourth keep going. As soon as you find it then you should get out and report, but if you do not find it, so, you have to go all the way and I did not find it then return minus one.

So, I should just scribble my logic, in a loop. So, here I would say well I start with my initial location which is i is equal to 0, then I say well. So, if a i is not equal to x , then I should go to the next one. So, I can just simply write if, keep going; else; return i you can just scribble. And then only you will start getting the idea, and so, if i greater than equal to n , so this is n minus 1, n is a size.

So, I can say n is equal to I am going to use n is length of a . If i is greater than or equal to n , return minus 1. So, this is logic. Now, after this I start thinking. Now let me write the code. So, this if keep going, I should put this is a loop. So, I will say I will just put these two in a loop. So, I can say; let us put it here. So, while; now this is then getting converted into a Python code, while $a[i] \neq x$. So, I should just increment i , in C you write like that i equal to i plus 1, and keep looping.

As soon as $A[i] = x$ it will come out anyway. So, this is the important part. So, this has been. So, this will give me the answer. I have to make sure that this is also true. Let us write the code, once the logic is there it is quite easy. So, let us start writing the code.

(Refer Slide Time: 22:59)



```
test.py No Selection
1
2
3
4 # Search x in an unsorted array A.
5 # Return index i, A[i]=x
6 # If x does not belong to A, then return -1
7 def search(A,x):
8     i = 0
9     while(A[i] != x):
10        i += 1
11    return i |
12
13
14
```

So, search x in an sorted unsorted array. Return index i, A i equal to x; if x does not belong to a this is nice when you come back to this code you know what this code does. So, it is written at the top. Now I will start writing the code. So, def name of the function. Now I will give you this 2 index, 2 arguments right. So, 2 arguments array a and x colon. Now I start with I equal to zero it is a good idea to keep some blanks reading is easier.

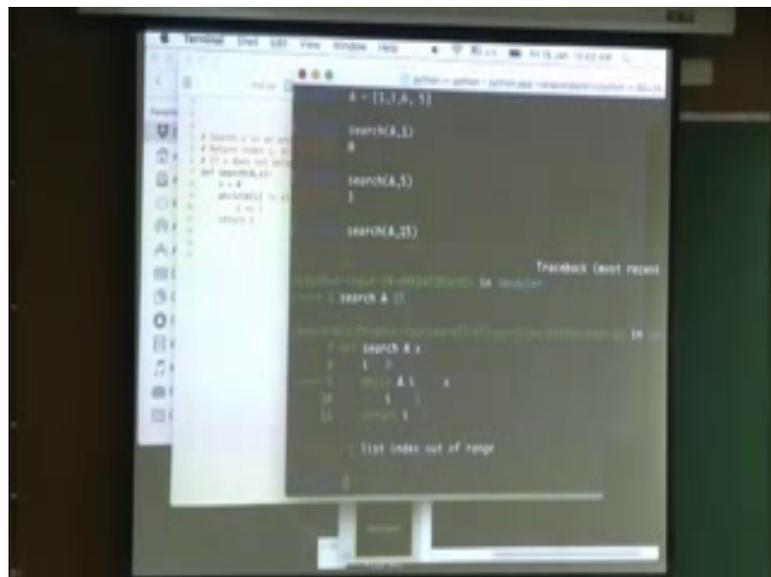
So, while loop while A i not equal to x colon i increment i. So, i is reserved for loop variable, good choice. Now as soon as x will come it comes out I can just say return i. So, In fact, return I; if x belongs to a; this should code should work. So, let us run it and check this 4 line code or 5 line code it works should work there is a problem if you know that if x does not belong to a then there will be a problem, but let us make sure it works. So, it is a good idea Python just it gives you a flexibility; does it work if it is a part of A. Then I will more worry about it; is simple line I have to include, but will do it right now.

But, let me first test. I should make sure I am in the correct directory. I am in the correct directory. So, run search. So, when I say run, it is going to import basically compile this function and keep it ready for my use. So, I can run. So, it runs without compilation error. Suppose I miss something you know. So, let us say I missed this colon no need to get Hasselt you have to just work. So, invalid syntax probably you are not able to see it also gives you line number. Actually I used it, so that the white is easily viewed from the screen.

So, it tells you where is the error? So, error is here. In fact, it is telling that invalid syntax. You can have a let us see probably not visible while it is in front of while. So, you know where to look for the error, and go there and so it runs. Now I need to check it works or not. So, I defined a array. So, let us do a list. So, list is as good here 1, 3, 6 unsorted so, I can put 5 here, 4 of them and so, I need to search. So, let us say I am searching in array a 1. So, I should get answer what 0 so, answer is coming 5.

So, it is working right 3 is coming. What if it is not in the list? So, it says the index out of range. So, it went there and it tried to do the competition.

(Refer Slide Time: 26:26)



So, I increased beyond so, it kept searching and it read it did not find it here it came here. So, I mean; I cannot search in the array now anymore it is beyond the size. So, this is the array it is giving. So, I need to fix it so, what I could say is so, I can just make a simple change. I will say well; if i equal to n this condition. I just insert in my code. So, if i greater than or equal to n break so come and shut the loop. Now return I will not be the correct answer no; actually it will give you a if I run it presently so, it would not give minus 1.

So, actually let us break. So, do not go to do anything in the loop; that is my suggestion. So, I have the index i, again based on the index i which comes out I can take a decision.

(Refer Slide Time: 27:43)

```
2
3
4 # Search x in an unsorted array A.
5 # Return index i, A[i]=x.
6 # If x does not belong to A, then return -1
7 def search(A,x):
8     n = len(A)
9     i = 0
10    while(A[i] != x):
11        i += 1
12        if (i >= n):
13            break
14
15    if (i < n):
16        return i
17    else:
18        return -1
19
20
```

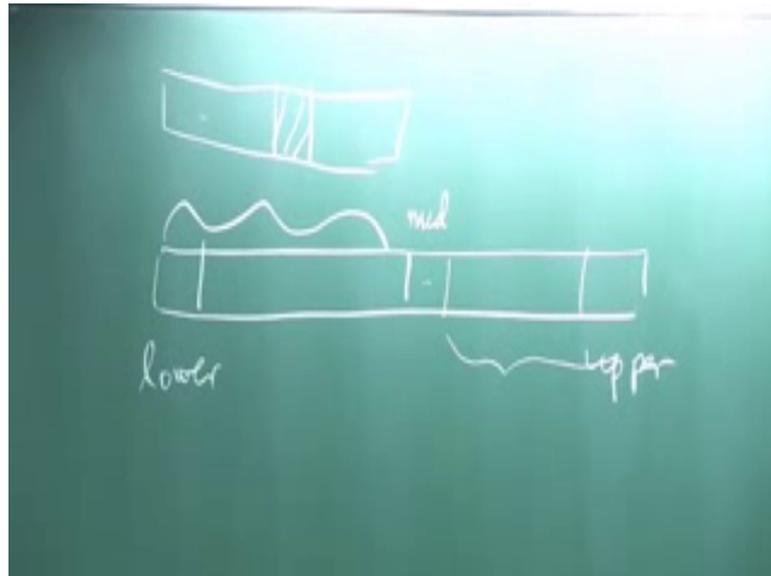
So, but other code is also equally equal to so, I mean I have no problem, but I i would like to write less if I greater than equal to n, then I know it is not part of the array; or let us say less than less than less than n then I know it is part of array right, but this code this loop can also return something outside, else: return minus 1. Let us again you have to again load it, because the code has changed so, run again. Now search n not defined. So, n is not defined. So, I can do n here, n equal to.

So, this will work. So, this how you write the code. So, it should be clean not tricks it is just simple loop, but each variable means something and will do it. Now this is for unsorted array how many operation does it require not always. So, you have to give for this problem. What is the best scenario, worst scenario and average scenario? The worst scenario is n you do the search everywhere worst case. The best case is 1 first time you got it and average is n by 2.

But suppose in a dictionary I want to search, dictionary is another search dictionary is really you search. How many words are there in a dictionary like 1000 page dictionary? There will be 1000 times at least 100 know. So, 100, 1000 words if you do linear search, you want to do linear search or do something else. Why dictionary is sorted a, b, c, d? Why is not jumbled up? So, is jumbled up not jumbled up because you can search by some clever scheme, and we do it.

In fact, I give the I ask the puzzle and everybody answered yes, yes; like I feel think a number between 1 and 100. Then how many now how many queries do you need to find out what the number I have thought about. Do you need hundred queries? You need less. How many; 6 actually 8. So, it is called binary search. So, the idea is , in fact; this code can be easily written. So, the idea is that it is sorted array.

(Refer Slide Time: 30:47)



So, sorting is a very important algorithm in computer science; and also in daily lives. You sort it for searching is faster; and how do I do fast search. So, if is my array is there, is increasing order. This is my lower variable and this is my upper variable. And we define a mid, which is average of the 2. So, see whether the number is less than mid or first equal to mid, equal to mid you got the answer. First is to check whether x is equal to a mid, then you return if you get it.

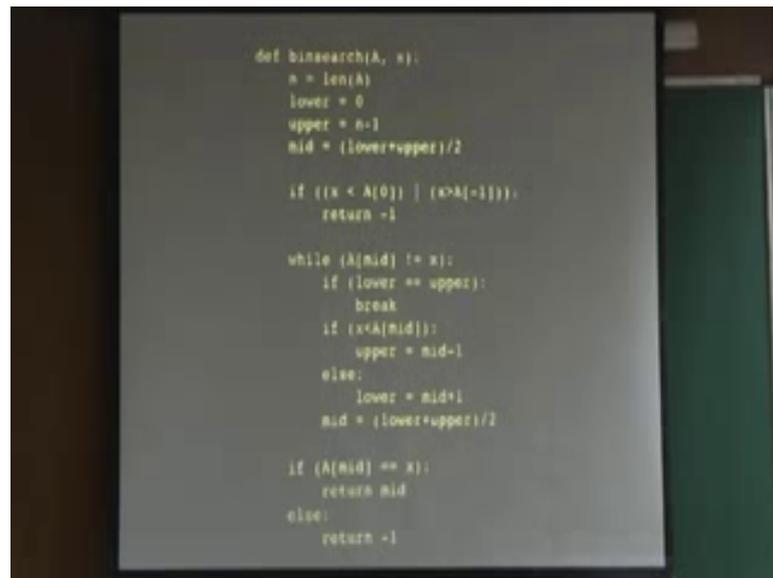
If you do not get it then see whether it is less than a mid, if it is less than a mid then where is the number is from here to here, but if it is greater than a mid then it is here. Now I do same thing for this part if it is here. Then again I go from, a sorted array this part. So, again I asked go to the mid; is it equal to this if not here or here; keep going down you converge quickly. In fact, you need $\log n$ steps if you think about it is a tree you creating each time you diving by 2.

So, to reach 100; I divide 50, then 25, then 12. So, that is how you keep going to searching and you reach in 6 steps. So, that is called log know log this is log. So, $\log n$ is

quicker. In fact, $\log n$ is real gain and that is why binary search is what is used for searching a sorted array.

And how do I write the code. Now I will not be able to write in 2 minutes. So, this was a standard one sorted array.

(Refer Slide Time: 32:53)



```
def binsearch(A, x):
    n = len(A)
    lower = 0
    upper = n-1
    mid = (lower+upper)//2

    if ((x < A[0]) | (x>A[-1])):
        return -1

    while (A[mid] != x):
        if (lower == upper):
            break
        if (x<A[mid]):
            upper = mid-1
        else:
            lower = mid+1
        mid = (lower+upper)//2

    if (A[mid] == x):
        return mid
    else:
        return -1
```

So, it looks this the idea is here, but the first check whether this array is, this number is within the bound. You sorted so it is easy, you should see whether it is less than a. So, this condition is important. You should write all possible scenarios; otherwise your code will break know. This is the condition if it is beyond the size. In the code, I have documentation some comments, but here I did not write the comment. Now, if it is so while it is not equal to a mid not equal to x, I shrink from the so, these are again a condition. If lower is equal to upper and a mid is not equal to x then what do you know; it does not belong to array right, a come down to the bottom of the tree and lower equal to it will be lower equal to upper at the bottom.

And if a is this is not true then you break. If it is less than a mid then you go to the left greater than a mid then you go to right. So, these are variables which will tell you what it means and you keep doing the loop. Now when you get out it could be this scenario; or it could be because, it was equal. Equal then I say give me the index of the mid, if it is not equal then I return minus 1 anyway. So, this is the algorithm and please think about it ok is not straight forward; it is not difficult either, but I suggest that you write your own

code, everybody all of you should write the own code and see whether you can do it alright

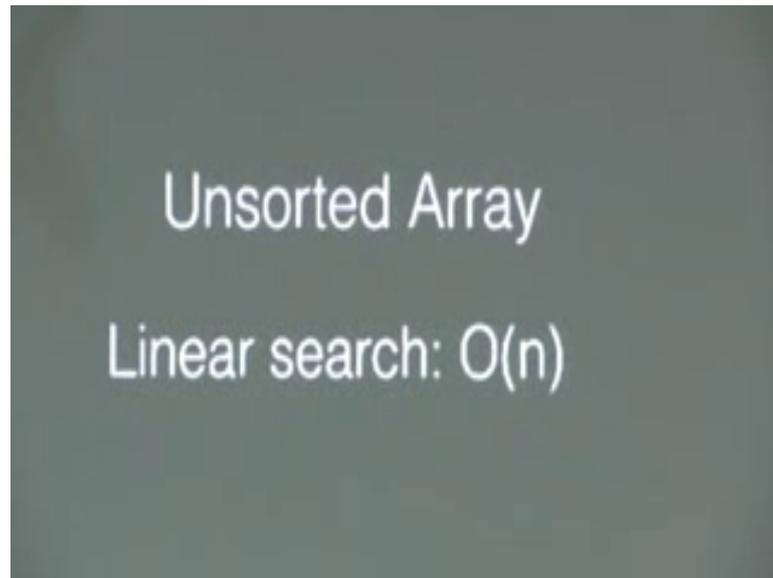
I do want to go bit slow; it does look easy programming, but it is not so easy; especially if you are not used to writing programs. I just want to make few remarks that computer is non intelligent entity. You may when there is another debate that computer is intelligent or not intelligent if you see some science fiction they will have their own views, but it is a programmer whose intelligent goes into the computer. So, you have to be you have to tell this computer in a very logical manner; extremely logical if you make a mistake it will not. So, and it has a way to understand.

So, you have to exactly tell a machine how to do you tell a machine you just have to make the very clear cut instructions oh, but it is also true this creator of C++ stroustrup you know you this book is very famous book C++ is a it brought in a big revolution in coding. So, he makes a remark which I do like it; he says if you can get it right on a computer, then you understood it completely. So, you may think that you understand some concept, but you can program and get the result, what is expected or what you think you predict then you got it right I mean that then you understood it completely.

In physics it is always recommended that you do the derivation yourself, do not look at the book and I understood it, but after that you should also program it if you can program it then you understand each step. So, for us when you simulate full flows you must have an idea it works this way, but when you simulate it then you get a better understanding. A programming is quite critical especially; as I said in modern times first class and it has so its own merits know you can understand it better. If you program it and get the result, in sometimes you are delving into unknown territory. So, I said it is a numerical experiment. So, the bottom line is you have to program in a very logical manner that is what programming is about.

So, I think what I want to do is? I just want to redo this just go quickly. I hurried up in the last class. I think it is important to emphasize some of the ideas once more. So, search algorithm will go through once more quickly now. So, first is called this order n, o is order n how long does it take or how many operations actually how long I cannot say because faster computer will do it in a shorter time.

(Refer Slide Time: 37:31)



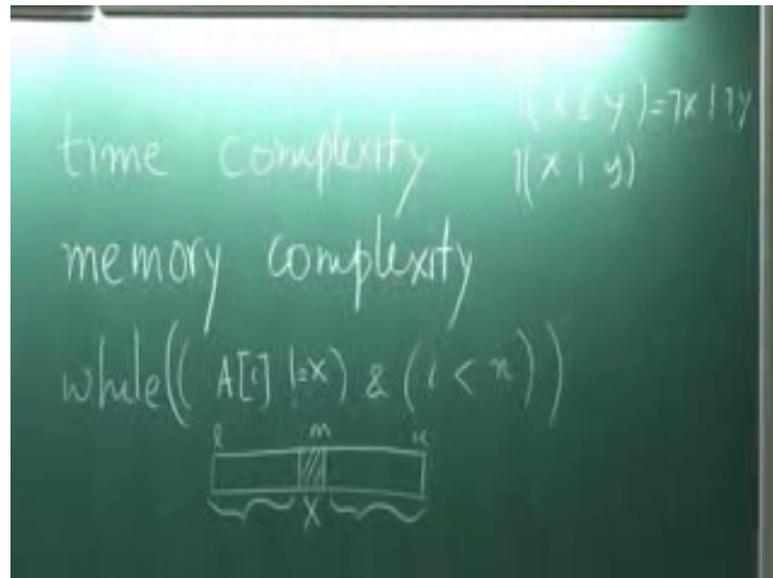
So, order means how many operations does it require, here operation will be comparison you compare 2 numbers. When I am searching in Google it is more sophisticated, but you are comparing strings or comparing so, Google of course, they money because they have very clever ideas they do not do like this they certainly do not do linear search, but comparison is important for this operation. For numerical computation you multiply, multiply and add and of course, divide.

So, division takes longer than multiplication. And multiplication takes longer than addition, but all of them are clubbed as floating point. Now their details how long does it take how, but each number is called float each operation is called floating point operation, multiply, add, division. Here, computations so how many operation does it take order n . So, we look at average. So, it is a set this; the fastest trace 1 because you get the 1 number. Slowest is n minus 1 were n actually in fact, the last number. Average will be n by 2. So, if I do 1000s of them, then I basically get number n by 2.

So, this is important if I will come to this idea again and again; numerical computation an algorithm so this is called algorithm. How do you search? I showed you in algorithm and this algorithm is linear search, in linear you go from beginning to the end and it has complexity this is also called complexity, I did not write the word complexity. So, this is called time complexity.

So, the 2 words may be you should know it. So, this is called time complexity.

(Refer Slide Time: 39:33)



So, number of operations and second is called memory complexity. How much memory does it take? So for this problem it take order n . So, numb array size is a memory required.

So, if you look at the code is very easy. So I did this in the last class. So, the while loop.

(Refer Slide Time: 40:09)

```
def search(A, x):  
    i = 0  
  
    while (A[i] != x):  
        i += 1  
        if (i == len(A)):  
            break  
  
    if (i < len(A)):  
        return i  
    else:  
        return -1
```

So, I do not want to use the for loop; because I may finish before going to the end, for requires beginning to the end. So, you do not need to go to the end. So, while A i is not equal to x. So, the index A i, well I am searching here. So, whenever I see i equal to x I

stop, but I keep going, until I do not get the condition right; not equal to, but there is a thing that if I is beyond this array I size then I should stop.

So, this is if condition and this loop this part within the loop. So, indentation is very critical that all these 2 sentences. In fact, well this is not a 1 sentence; well this is 1 sentence sorry this is 1 sentence this 1, but there is a block consist of 2 statements. So, these 3 together are within the while loop. Now after I get out of the so once I finish here then I come out of the loop. This is obvious, but you must remember you must come out of the loop. Now there are 2 ways, I can come out of the loop; either by this break condition or by satisfying this satisfying opposite of this condition.

So, you can satisfy opposite of this condition; that means, i is less than length A right. So, i is a value of your index, but if this condition is satisfied; that means, the x does not x is not inside the array A and I should return minus 1. So, this function returns a number integer either the index of array a where x is present or minus 1 or I could have written this while loop in slightly different way, if I want to avoid break.

So, the way to do is I want to put both the conditions in the while. How should I put both the conditions in the while? So, I have to say that of course, $A[i] \neq x$ and i is less than n . So, both if both the conditions are satisfied you continue; if any of the condition is wrong, then you get out.

Student: (Refer Time: 42:43)

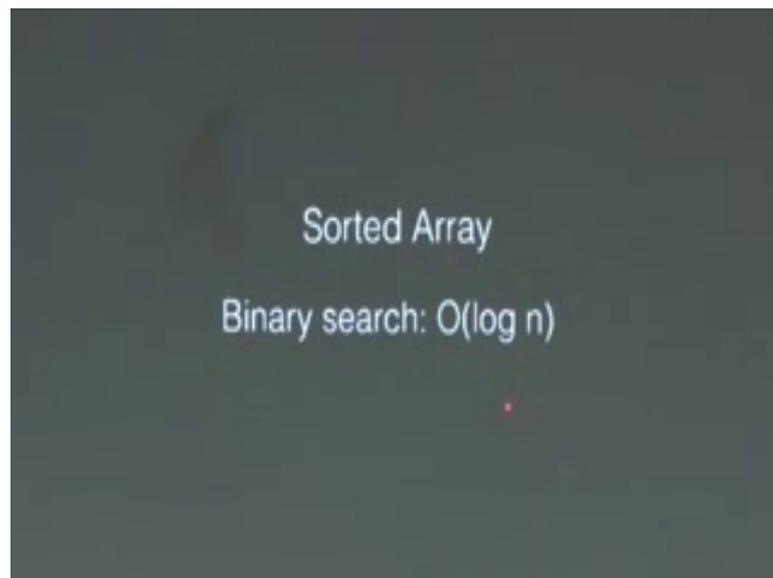
So, it will look only one the first time x encounters it will come out. So, this program will give you only one index not 2 index good question it gives you only 1 index. So, if I put so if I want to avoid break then I have to say $A[i] \neq x$. So, if this is true and I do get mixed up whether should I put r or should I put n . So, that is why break is easy to think of. So, this can be confusing sometimes and my mistakes are there in this either AND or OR.

So, I can tell you that much, i less than n no i so I continue till these 2 conditions are satisfying. If any of them when will it get out, if any of them is not satisfied? So, if this is not satisfied it will get out or this is not satisfied this is get out. So, Boolean stuff you should also, x AND y x OR y not of this is what. So, this is not x OR not y . So, this I am saying is not. So, this one is important when you program. In fact, you need it what about

not of this, this will become AND. So, you have to do this. So, this is one place where people do tend to make mistakes.

Now as I said, this code is not efficient. I can do better; if I do some more work on the array. In dictionary we somebody has to compile it in a sorted manner it has been sorted. We can have an array which is sorted, then how long does it take $\log n$ ok.

(Refer Slide Time: 44:56)



Is this proof clear to everyone, yes or anyone has problem with understanding $\log n$, you go faster. So divide in half divide in half. So, if I search for a number within zero to hundred, then I look at in the first half or second half then if it is in the first half then again I divide in 2, so you reach in $\log n$ steps. So, complexity here is $\log n$. So, time complexity will be $\log n$, $\log n$ is very quick you can plot it $\log n$ is quick

Student: \log base two

Yes this is \log base two indeed. There is one more when I say order; order means that factor I am not worried about in front, so the factor in front of so, it could be if I say \log base 10 then there will be factor in front. So, of course, the people who write codes are worried about the factor. Factor 2 is better than factor 3. And it saves you lot of time. If you go from 3 to 2 you have saved 33 percent of the time. Now, let us do the recursive one first. This is nice actually recursive one.

(Refer Slide Time: 45:57)

```
def binsearch_recursive(A, lower, upper, x):
    mid = (lower+upper)/2
    if ((x < A[lower]) | (x>A[upper])):
        return -1
    elif ((lower==upper) & (A[mid] != x)):
        return -1
    else:
        if (A[mid] == x):
            ans = mid
        elif (x < A[mid]):
            ans = binsearch
                _recursive(A, lower, mid-1, x)
        else:
            ans = binsearch
                _recursive(A, mid+1, upper, x)
    return ans
```

So, What I had said? That first this is the condition, when it is beyond the array is the so it is sorted. If it is less than first element and above the last element, then it is not part of the array right it is sorted. So, you can just do this. Now if is the if it is not true, then I am going to else if. So, if this is not satisfied then I come here. Now I will like to search, now please remember that here I am searching from lower to upper; this index is important.

Now this has been not searching the whole array, it is searching from lower to upper. Now what I do is, I see whether lower equal to upper; that means, it is only 1 element and then if a mid is not equal to x so I am putting and so; that means, x does not belong to array a from lower to upper which is equal for this condition. So, again I return minus one. Now if none of this condition is true, these 2 conditions; then I come here else. Now, what all possibility? Now lower is not equal to upper or lower equal to upper, but this condition is not satisfied.

So, you have to logically argue as I said computer will under you have to put all your logic at hand. So, now, this condition this is not of this will give you this so A mid equal to x. So, it will by the way it should return a number it should return either index or minus 1. So, this will give you a index or minus 1. So, this will give this is the answer. If a mid equal to x, then I got the answer.

So, this will be there when lower equal to upper. Now of course, you can put more else if is possible that you can put this condition, if this equal and equal to. So, you could be a part of it. If you like to make the code simpler is good is no problem. Now if this is also not true, this is not true; then lower is not equal to upper and A mid is not equal to x. So, I had made this diagram. So, this is mid so, this is lower and this is upper. Now so far by doing 3 conditions I exhausted x is not here and lower is not equal to upper; that means, it is either here or here.

Now you put this condition, if x is less than A mid; that means, it is in the lower half then I should do the binary search in which range, lower to mid minus 1. And this will give you an answer it should give you an answer should not print I want an answer. So, if you print then it is this is not what I want. So, it will give you an answer. So, this sentence continues and I put it line break, but if this is also not true, then x is lying outside; right side of m so, from here till here.

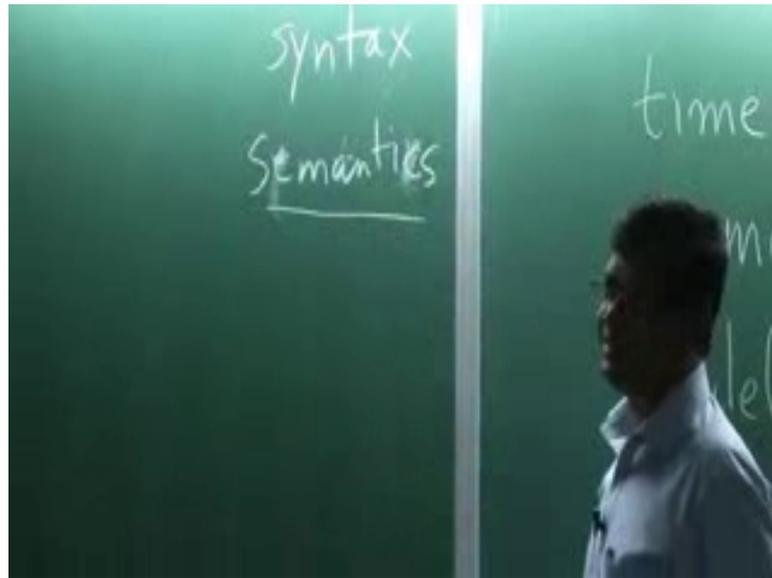
So, I do binary search from mid plus 1 to upper. And this so answer will be either this 1 of the 3, and I should return. So, this part all of it all these is part of else and all of it belongs to the function. So, indentation is very critical. So, this for Python in forces indentation in C will not stop in C code you could put everything in line 0 or column 0, but that is not a good practice. So, you should have a indentation in your code. Well I mean; this is how you have to write how to search x.

But in x while executing what will you do. So, you will look for let us say x is existence side. So, it will find middle and search a look for compare this number, if it is not equal then either it will go to the left or right depending on that condition. Now if it is in the left then, again it is going to create another mid, m prime. Now we will see whether this x is equal to m prime, if it is not then it is going to go further down either here or here. It is not well now let us say it starts here again it is going to go down.

So, this is going to do recursively and once it finishes that is where the answer will be. And of course, it will top up so from bottom it is going to come up so the same answer is transmitted. That is how this binary search works. It is a recursive procedure, function is using itself. I would recommend that you can print answer. So, just take as array which is not very large and keep printing.

One way to debug your code is to remove the errors. So, first compilation error, you will get some error this syntax is not right; syntax is grammar, your grammar is not right it will not even understand, but if grammar is right, but then I do not understand what you are saying your English is perfect, but I do not understand what you are saying or you are not doing what you are not saying it correctly the semantics that is semantics. So, in computer science whether two special words syntax.

(Refer Slide Time: 52:34)

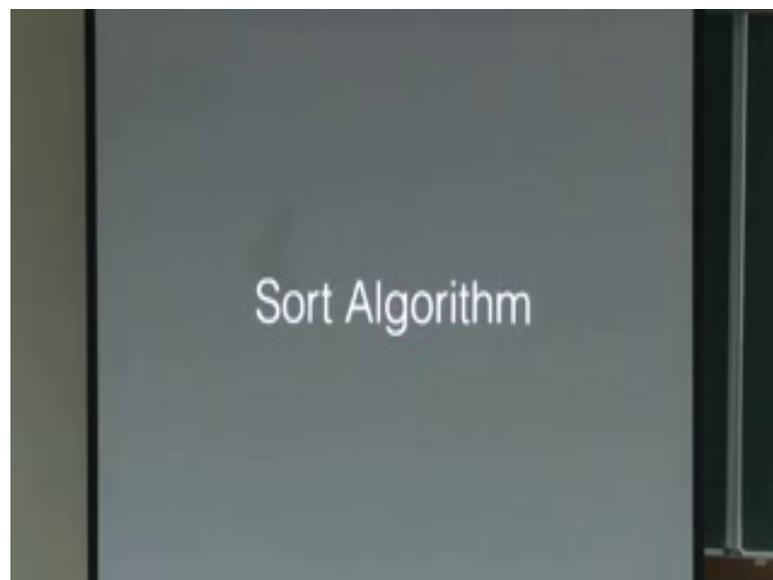


So, this is comes out while interpreting. So, error comes know function not defined all that error comes. Other one is semantics. My English is not good semantics se semantics es I think. So this is meaning. So, what is the meaning of your. So, if your semantics is not right this code will not give you the correct answer. So, that is why you need to test your code. So we when we have to test your code is put log print. So, one idea is to just print here before return. So, you see output what is happening at different loops. This code is going to be expensive. This code takes longer time because it goes down and comes up. So other idea is, to do in a non recursive manner. So, no recursion you do it non-recursively; that means, I do with loops.

So, this is same code. In fact, recursion is you get my idea right. In fact, I did recursion first, then you can do non-recursion. So, this code is also not very long. So, basically this is the first condition. Now this part, I am using break for this second condition, lower and this is basically either goes to the left search or right search then it keeps looping. So,

they exactly doing what I did for recursion, but except it is non recursion either choose left or right left right keeps going down and i so, I would say that put some print in this code and see the range. How long? So, I am staying $\log n$. So, you can show yourself or you can convince yourself, that it is indeed $\log n$ alright. So, it is good idea to print and check. Here again this condition is required. Now so I assume that I have sorted array. So, if the array is not sorted. So, one important point is to do sorting so. In fact, when you do fill up the excel sheet it is sorted in role number wise, know it is easier to search full in the large class.

Refer Slide Time: 55:02)



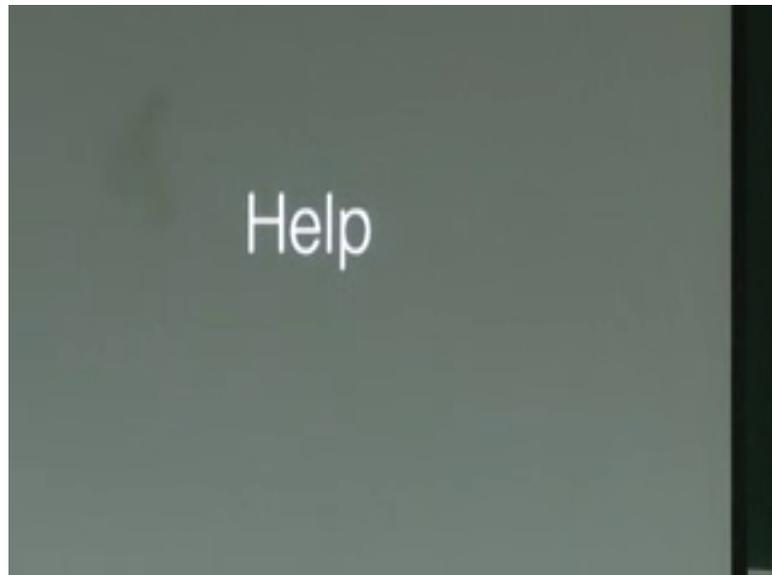
This algorithm is a big industry for sorting a data. Numbers are of course, easy to understand you sort in terms of increasing order or decreasing order, but you also sort strings. So a, b, c, d there is a way to say which lower and which is higher. Sorting homework so please do it. Sorting is easy for the simplest algorithm. Of course, there is large number of sorting algorithms the easiest 1 what will be the easiest 1 will be easiest sorting algorithm. I have to keep the number in decreasing order let us say bubble sort.

So, first pick first get the largest number by searching. So, you search and pick the largest number you can do binary search if you like put first. Then remove that number; largest number is here. Now from number 2 onward to n the last number I search for the second largest put it here. So, I keep doing it and that how that is how I will sort it yes it is not a. In fact, it is the worst algorithm. How long with the how many operations will it

take? Sorry. First time it will take n , second time it will take n minus 1. So no so you can just do it if I just do it, n plus n minus 1 plus 1.

So 1 to n you sum it up n into n plus 1 by 2. So, it is order n square. So, it is order n square algorithm. Sorting can be done in $l \log n$ the good ones so. In fact, you can bring in some of these ideas for, so this is called divide and conquer. So, divide the data in two parts do the first half second half. So, will not do that part sorting with more complex algorithm, will save time, but that is not this again is not part of the course really. Now the last thing is help.

(Refer Slide Time: 57:26)



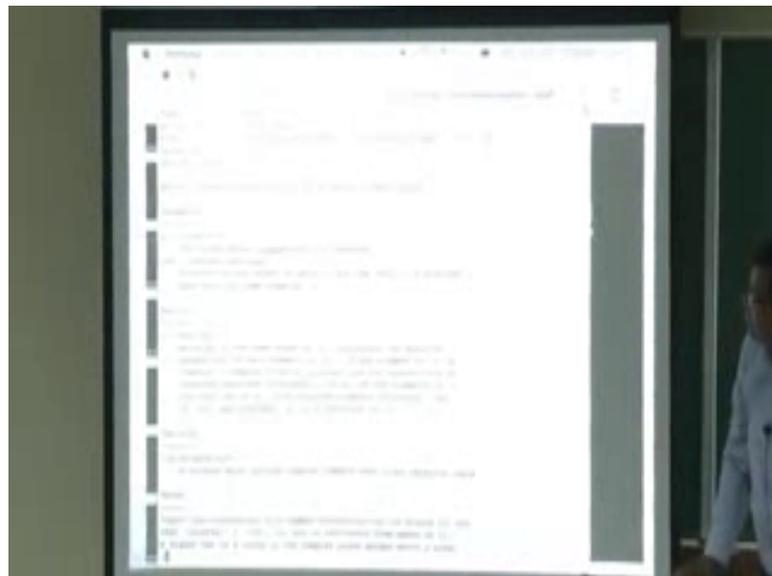
So, Python has very good online help, when it will be part of your I Python. So, I can go to the terminal and I do it.

(Refer Slide Time: 57:34)

```
In [1]: ? plot
In [2]: ? sqrt
In [3]: math. TAB
In [6]: help()
Welcome to Python 2.7! This is the online
utility.
help> array
help> list
help> matplotlib
help> quit
```

But I will just tell you from here it will save time. So, if you put a question mark people are using it. If you put a question mark and the function name, you will get details about it. Let us just do it quickly. So, question mark. Suppose I want to know what does square root do? So, it tells you what is square root? So, square root.

(Refer Slide Time: 57:52)

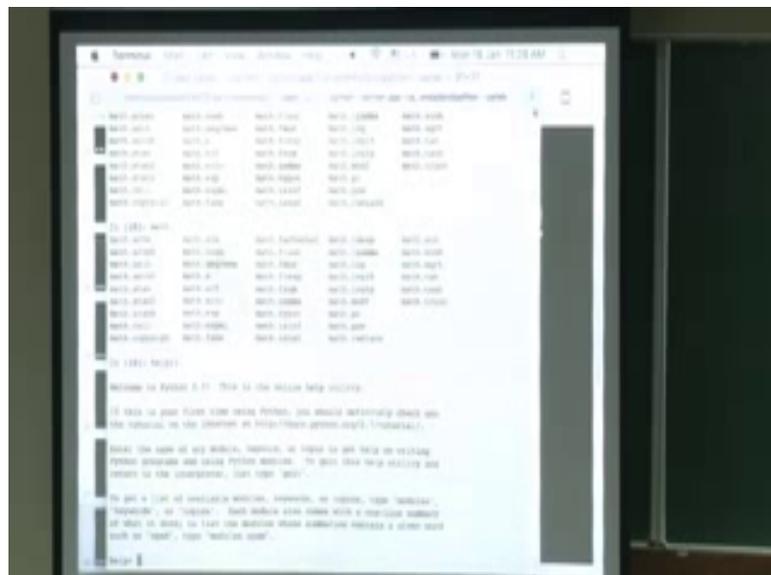


It gives the details about square root function. Now I can just quit. So q is quite. You can so plot I am looking for some symbol you will get some discussion about it. So, it is very useful, I do not need to go to internet; this is part of Python I Python. Now the one

sophisticate thing that within a math library, so import math know, but I want to know what all math library does. So, math I think is imported already. So here math dot please see put a tab, tab everybody knows tab. This put a tab then you will get all the libraries of math. Math tab math dot tab. So, I am just putting a tab. So, it just gives. So, is very useful if you are looking for, I mean; suppose I forget the syntax of log or log 10. So, you get it from here.

So, now there is another thing called help. Online help this I believe will require internet. So, if I just say press help colon this thing. So, it I can now get help on the module.

(Refer Slide Time: 59:20)



Here, I will not online help the help I just showed you right now, before this you will not tell you about array. What does object do? What does a module do? But here if I just say array, it tells you all about, is a module array. So, it is more on module and libraries. So, matplotlib if you want to know. It is tells you that it is a package, matplotlib it is the name of the package, what is the file? So description so, it is very detailed online help.

So, any question on this. So, basically I finished programming. So, it was quick right I mean if I did C, it should be taking double the time. Be careful with function. It is very powerful, but it can be frustrating, I did get frustrated when errors come not able to get the answer, but Python has evolved is easy, but you have to be careful. Python returns any type. So, it is a def function name. So, it does not tell you what type. So, it could return integer, it could return float, it could return an array itself or it can return 2 objects.

So, it has suppose I want 2 indexes for example, you are saying I know this x is re x is occurring twice. So, I could get index i 1, i 2. So, when you say return, you have to put i 1, comma i 2. So, it will return two of them and where will that be. So, I could so the answer has to be, to invoke it you have to say x is equal to x 1 comma y 1 equal to search A comma x.

So, first one will be stored in x 1 and second one stored in y 1. So, if you are expecting something, you better store it somewhere otherwise; the function is not going to give the answer. So, you should store it. So, that is why Python is a cryptic, in the sense it does it fast, but you must be careful what you are getting. When you say print in a function and you know not returning anything, then you do not need this. This part is not required otherwise; you need it. So, this part is programming part.