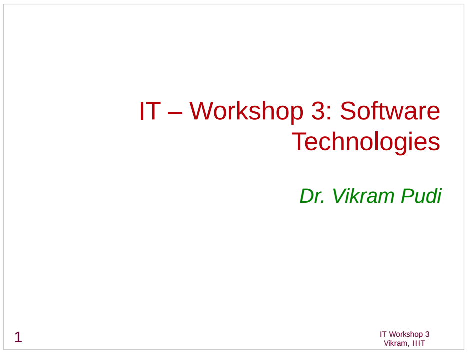
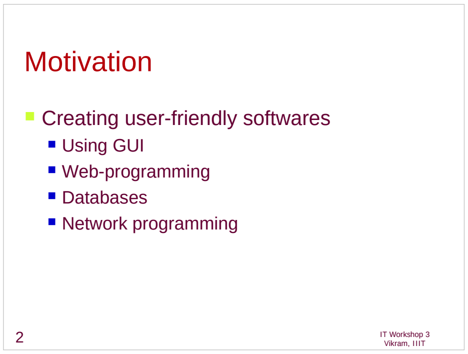


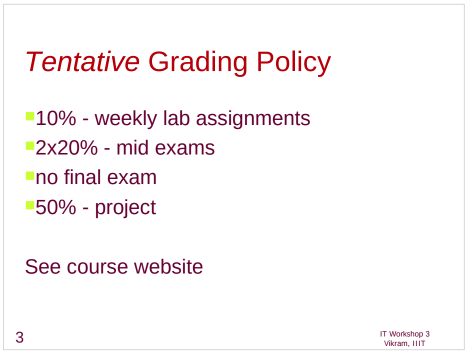
IT – Workshop 3: Software Technologies

Dr. Vikram Pudi



Motivation

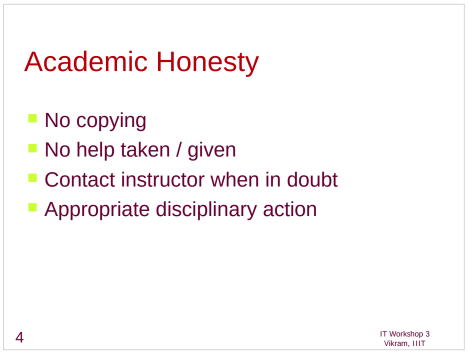
- Creating user-friendly softwares
  - Using GUI
  - Web-programming
  - Databases
  - Network programming



Tentative Grading Policy

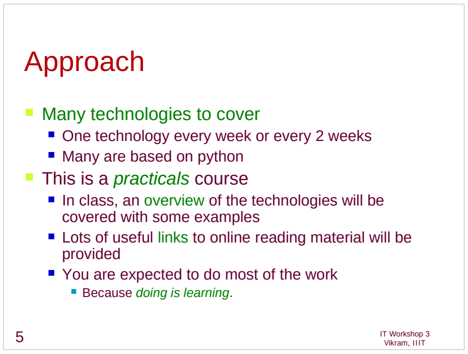
- 10% - weekly lab assignments
- 2x20% - mid exams
- no final exam
- 50% - project

See course website



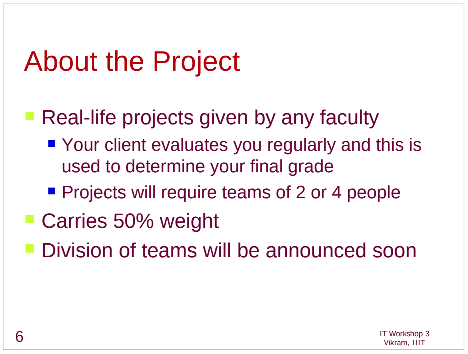
Academic Honesty

- No copying
- No help taken / given
- Contact instructor when in doubt
- Appropriate disciplinary action



Approach

- Many technologies to cover
  - One technology every week or every 2 weeks
  - Many are based on python
- This is a *practicals* course
  - In class, an overview of the technologies will be covered with some examples
  - Lots of useful links to online reading material will be provided
  - You are expected to do most of the work
    - Because *doing is learning*.



About the Project

- Real-life projects given by any faculty
  - Your client evaluates you regularly and this is used to determine your final grade
  - Projects will require teams of 2 or 4 people
- Carries 50% weight
- Division of teams will be announced soon

Time-table

- Aug 2: Projects begin
- Aug 9: DB design deadline
- Aug 23: Screenshots deadline
- Sep 6: Implement screenshots deadline
- Sep 20: Connect to DB deadline
- Oct 4: Working prototype deadline
- Oct 11: Refined version deadline
- Oct 18: Report deadline

Python Review

Topics

- Data types: numbers, strings, lists, dictionaries, tuples, files
- Control: if, while, for, functions, generators
- Modules: sys, os, shelve, math, re
- Advanced: Functional & OO programming

Problem 1

Argument passing mechanism in python?

Names are passed using call by value, values are passed using call by reference.

```
def func(s):
    s = [4,5,6]
s = [1,2,3]
func(s)
print s
o/p: [1,2,3]
```

```
def func(s):
    s[0] = 4
s = [1,2,3]
func(s)
print s
o/p: [4,2,3]
```

Problem 2

```
def union(x,y):
  z = [ ]
  for i in x:
     if i not in z:
          z.append(i)
  for i in y:
     if i not in z:
          z.append(i)
  return z

print union(“abcd”, (‘e’,‘f’,‘g’,‘h’))

#OO-Feature: polymorphism
```

Problem 3

Consider an nxn chess board where each row has two pawns at locations specified by a list of tuples named positions. Each tuple in positions is of length 2, specifying the two positions in each row where the two pawns are located.

```
def prettyprint(positions):
   for i in positions:
      print ‘.’ * i[0] + ‘X’ + ‘.’ *
        (i[1]–i[0]–1)+‘X’ + ‘.’*(n–i[1]+1)
```

Problem 4

Consider a binary tree.

(a)Define an node class in python to encapsulate a node in a binary tree. Each node has 3 fields: data, leftchild and rightchild. Write a suitable constructor to supply the data and create dummy left and right children initialized to 0.

```
class Node:
     def __init__(self, data):
          self.data = data
          self.leftchild = 0
          self.rightchild = 0
```

Problem 4 (contd)

(b) Define a method for the Node class named makeLeft that creates the left child node containing a data field initialized to 0. The method should return the new node created.

```
def makeLeft(self):
   self.leftchild = Node(0)
   return self.leftchild
```

Problem 4 (contd)

(c)Define a method named show for the Node class that prints the data field of a node and of all its children in an in-order fashion.

```
def show(self):
   if self.leftchild:
        self.leftchild.show()
   print self.data
   if self.rightchild:
        self.rightchild.show()
```

Lab Assignment for Week 1

- Implement a binary search tree data structure in python.
- Write methods to insert nodes, delete nodes. Also write methods for in-order and pre-order traversals. Use generators for the traversals. Finally write a method to search for a given element in the search tree.
- The main program should take 2 lists of numbers as input. The first list is used to populate the tree. The second list of numbers should be searched for in the tree.
- The output is 3 lists. The first list contains zeros and ones depending on the search results. The second list contains the in-order traversal of the tree and the third list contains the pre-order traversal.

Example input and output

i/p: [2,5,4,9,11] [4,2,1]

o/p: [1,1,0] [inorder-traversal sequence]
[preorder-traversal sequence]

Follow this format as your assignment will be evaluated automatically.

Assignment 2 (Due: 1st Lab class)

- Write a program that goes to all the faculty home pages linked from: http://www.iiit.ac.in/faculty/faculty.php
- Create a shelve that has the faculty name as key and value is a tuple containing: (Areas of interest, email address, phone number)