07/12/22

WEDNESDAY

03 APRIL 2013

APRIL 2013

| M | 1 | 8 | 15 | 22 | 29 |
|---|---|---|---|---|---|
| T | 2 | 9 | 16 | 23 | 30 |
| W | 3 | 10 | 17 | 24 | |
| T | 4 | 11 | 18 | 25 | |
| F | 5 | 12 | 19 | 26 | |
| S | 6 | 13 | 20 | 27 | |
| S | 7 | 14 | 21 | 28 | |

6th sem:

Sess: 1 (SE)

*. Find various types of software:-
(Real time Ex: ?).

- ↳ photosob
- ↳ 4K video downler
- ↳ Eclips
- ↳ VS code.
- ↳ Database.

use.

*. Requirement
↓
Parameters
Choose (Tech.) → Design
↓
Develop.

To build a software.

* Book:- Fundamental of Soft. Eng.
Author: Rajib.

Meetings | Things to do | Important Calls

THURSDAY

04

2013
APRIL

(Ch: 1, 2, 4, 7.) — S.I.
↓
by others

APRIL 2013

| M | 1 | 8 | 15 | 22 | 29 |
|---|---|---|---|---|---|
| T | 2 | 9 | 16 | 23 | 30 |
| W | 3 | 10 | 17 | 24 | |
| T | 4 | 11 | 18 | 25 | |
| F | 5 | 12 | 19 | 26 | |
| S | 6 | 13 | 20 | 27 | |
| S | 7 | 14 | 21 | 28 | |

* cot is planing ?
* cot is meaning of feasibility ?
* Stock & inventory management Sys.
* Find out Project def" !!!
* cot is driver ?

FRIDAY

05

2013
APRIL

V. (Nov)

AJT

APRIL 2013

| M | 1 | 8 | 15 | 22 | 29 |
|---|---|---|---|---|---|
| T | 2 | 9 | 16 | 23 | 30 |
| W | 3 | 10 | 17 | 24 | |
| T | 4 | 11 | 18 | 25 | |
| F | 5 | 12 | 19 | 26 | |
| S | 6 | 13 | 20 | 27 | |
| S | 7 | 14 | 21 | 28 | |

* J2SE :- Java 2 Standard Edition.
* J2EE :- Java 2 Enterprise edition. (Use for web design)
* J2ME : Java 2 Micro Edition
* .jar : collection of dot class files.
  (java archive. compress)
* .war : in web project.
  (web archive
* .ear : in Enterprise application.
  Enterprise archive.
* 2 kind of Servers:-
  app" severs :- to deploy .ear file.
  web servers :- to deploy .war file.
  ⇒ It has internally support of web servers.
* for designing version :- 1.6 ] versions.
* netbeans :- 8.2

Meetings | Things to do | Important Calls

SATURDAY 06. 2013 APRIL

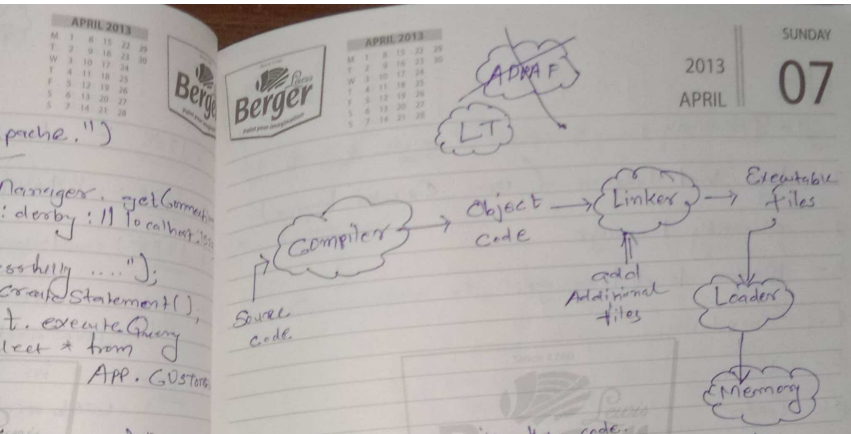
```
try{ Class.forName ("org.apache.")

Connection cn = DriverManager.getConnection
        ("jdbc:derby://localhost:1...

S.o.p("Connection Successfully ...");
Statement stmt = cn.createStatement();
ResultSet rs = stmt.executeQuery
        ("select * from
         App.Custom...

while (rs.next()){
    getString("colName"); getFloat(
    S.o.p ("Customer Name : "+
        rs.getString("Name");
    st.close(); con.close();
}
catch( Exception e) { ... }.
```

SUNDAY 07 2013 APRIL




* Compilers :- Compile the code from source to objcode (.obj files).

* Linker :- Links the code with some additional files. (Eg: Header files) and convert it into executable form. (.exe file).

* Loader :- Loader loads the (.exe) contents in to the memory.

08 2013 APRIL

* Definition:-

1- High level language (User Oriented language)
2- assembly language
3- Machine level language
4- Translators.
5- Linker
6- Loader

#C 8 marks fixed..)

* Real time applications of Lang. translator.?
↳ Health Care Industries.

* Jeffrey D. Ullman (70's-80's)
* BS Publication. (80's-90's)
} Book

* Answers:-

1.A. Human can understand.

2.A. Microprocessor can understand.

3.A. Machine (Computer) can understand.

4.A. Translator which can translate the one lang to other lang

TUESDAY 09 2013 APRIL

5.A & 6.A. Check:- previous & previous page.

* Computer:-

↓

Combination of H/w parts + S/w

↓ piece of mechanical devices } 1. H/W

↓ it is fun are being controlled by compatible. S/W } 1. S/W

H/W
1.
2. Understand instruction in form of Electronic charge.

S/W
1.
2. Which is counter part of binary Lang. in S/W.

Meetings | Things to do | Important Calls

8/12/22.

FRIDAY 19 APRIL 2013

LT

## Ch:1 Language Translation Overview:-

1). NIL & Component of System.

(i) Application software :-

(a.) word
- Bank management sys.
- Telegram.
- Whatshapp - Desktop.
- Accounting s/w

→ System s/w :-

(b.) - O.S. (Windows, mac OS, Linux.).
- Compiler
- assembler
- interpreter.

* Cimp #2m] :-

fig:2 - System s/w is a barrier betn hardware & application s/w.

Usage of sys. s/w :-
- Because without sys. s/w we can't run app. s/w.
- sys. s/w used to high level to low level lang.
- to store

Meetings | Things to do | Important Calls

→ Manage basic functions such as storing data, retriving files and sch... task etc.

→ To make execution of each pr...

→ To make furtive utilization of all the resources.

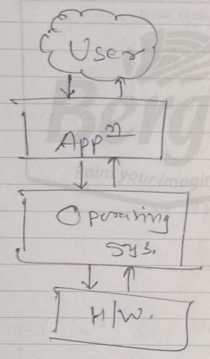



fig:1

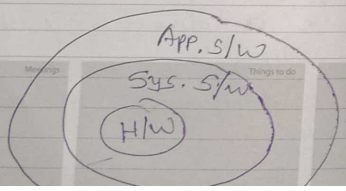



fig:2



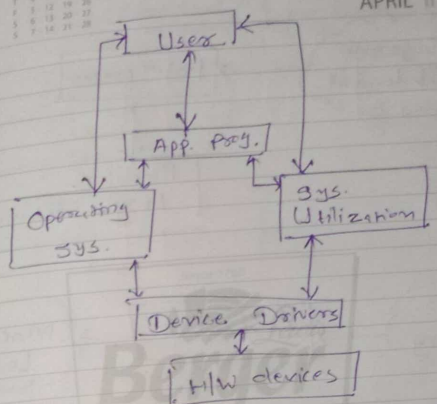
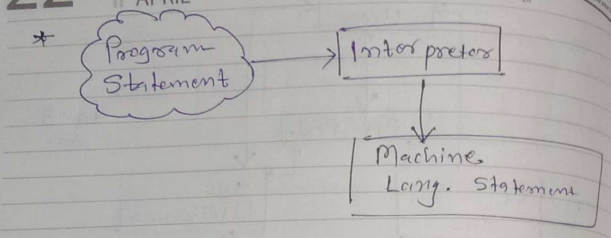


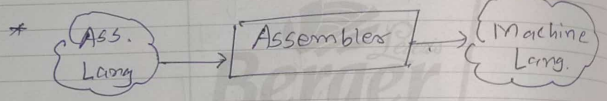

fig:3

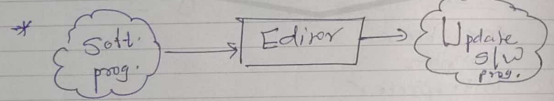
* All Components of Entire sys. (fig:3).

1. Assembler
2. Compiler
3. Interspreter
4. Editor (To update the S/W)
5. Linker
6. Loader.
7. debuggers
8. Macro
9. O.S.
10. Device Drivers.

MONDAY 22 2013 APRIL



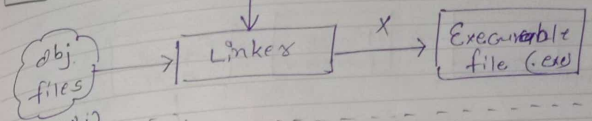



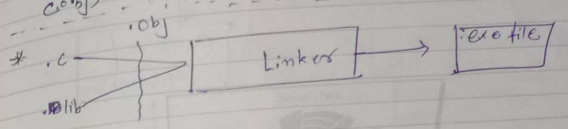

⇒ Types of Editors :-

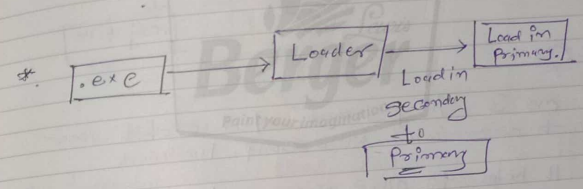
1. Line Editor
2. Screen Editor
3. Word processor
4. Structure Editor.

TUESDAY 23 2013 APRIL

Additional files. (Library files)



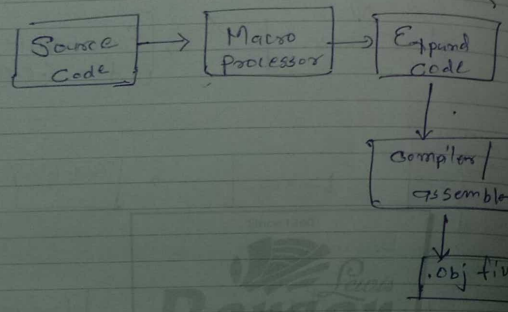




* Debuggers : it is a computer program which is used to find out the errors which are also called as 'bugs' in source prog.

* Macro : Code reusability.

WEDNESDAY
24
2013
APRIL




* OS :-

- OS is s/w programe that enable user to communicate with comp. Hardware
- It helps to other program to run & control them.

Ex: Linux, window, Mac OS

- Single use
- multitasking
- Multi user
- Networks
- distributed

} Types of OS.

THURSDAY
25
2013
APRIL

* device drivers :-

- it is a sys. program.
- Used to control no. of devices, which are attach to Computer.
- device drivers tell to O.S that how the device will work or certain commands which are generated by the user.

Ex: - Mouse drivers
- keyboard "
- Wi-fi "
- Bluetooth "

* Language Processes Activities :-

↳ Language Processor :-

⇒ Convert source code into Machine Code.

⇒ the activities which are Processed by Compiler.

FRIDAY

26

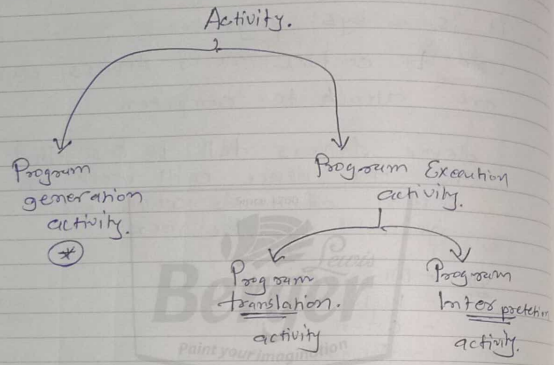
2013

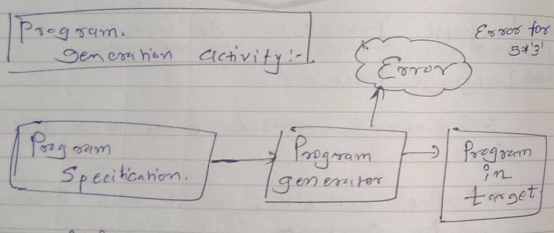
APRIL

APRIL 2013

M 1 8 15 22 29
T 2 9 16 23 30
W 3 10 17 24
T 4 11 18 25
F 5 12 19 26
S 6 13 20 27
S 7 14 21 28

⇒ Types of activity:-




- 2+2
- 5*3

- 4

Design Pattern (FMV)

2013 APRIL 27

SATURDAY

Sess:-1

or:→ HTML, CSS, JS, Bootstrap } FMV.

⇒ a design patterns are well proved sol^n for solving specific prob^m / task.

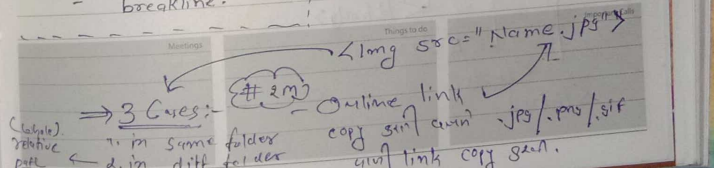
* Creational Pattern
* Structural Pattern.
* Behavioural Pattern.

Design.

* Basics tags of HTML:-
  1. <html> </html>
  2. <head> </head>
  3. <body> </body>.

* Text Formatting tags:-
  - <font> #imp
  - heading
  - para.
  - breakline.

- Ex. of tags v/s attribute.
- hyperlink tag:-
  <a href="link"> ... </a>.



SUNDAY

28

2013
APRIL

APRIL 2013

| M | 1 | 8 | 15 | 22 | 29 |
|---|---|---|---|---|---|
| T | 2 | 9 | 16 | 23 | 30 |
| W | 3 | 10 | 17 | 24 | |
| T | 4 | 11 | 18 | 25 | |
| F | 5 | 12 | 19 | 26 | |
| S | 6 | 13 | 20 | 27 | |
| S | 7 | 14 | 21 | 28 | |

* Create an HTML page having table which contains Eid, EName, E-sal, E-ContactNo fill out 5-details. #6m

*=>

```
<html> <head> </head>
  <body>
    <table>
      <thead>
        <th> Eid </th>
        <th> EName </th>
        <th> E-sal </th>
        <th> E-ContactNo </th>
      </thead>

      <tbody>
        <tr>
          <td> 101 </td>
          <td> ACB </td>
          <td> 20000 </td>
          <td> 9876 xxxx </td>
        <tr>
          : } </tr tags>
      </tbody>
    </table>
  </body>
</html>
```

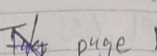
Meetings | Things to do | Important Calls

MONDAY

2013
APRIL

29

* List tags :-



1. <ul> </ul>
2. <ol> </ol>

↳ Type = "1" : (arabic numbers)
"A" : (Uppercase alphabets)
"a" : (Lowercase " )
"I" : (Roman Uppercase alpha.)
"i" : ( " Lowercase " ).

* Form tags :- <form> </form>

↳ Container type

→ <Input type="..." value="..." />

* Create a Reg. form :- #6m

→ f.name, l.name, m.name, email, username, pwd, gender, hobby, clear btn, submit btn.

Meetings | Things to do | Important Calls

Appointment sys. {PHP + JS + Front-end.

DSP [FMV]

*. 9/12/22 NOTES

* JS: light weight integrated pro. lang.
* It's integrated with HTML & JAVA
* It can't be implement as a whole language But it can be implement as a part of webpage.

*. Client side javascript :-

* Advantages :-

1. Less server interaction.
2. Immediate feedback to the visitors.
3. Increased interactivity.
4. Richer Interface.

* Limitations :-

#2 Marks → 1. javascript advantages & limitations.
(fixed)

2. javascript development tools.
3. syntax of js.

~~Scripto language - Javascript~~

# Var, let, const

<!-- contents -->

HTML comment

NOTES

* <script language="Javascript"
type="text/javascript">

// code

</script>

(fixed) & Marks: *

<noscript> ... </noscript>

↳ In some web browser doesn't support js. then <noscript> </noscript> will execute (if...else vaj)

---

1. Var :-

→ var name = "vraj"; ✓
var name = 10; ✓
var name = true; ✓

refer "my PPT".

2. let :-

let name = "vraj"; ✓
let name = 10; ✗
name

3. Const :-

const name =

* Semicolon is optional.
Comment is optional.

Meetings | Things to do | Important Calls

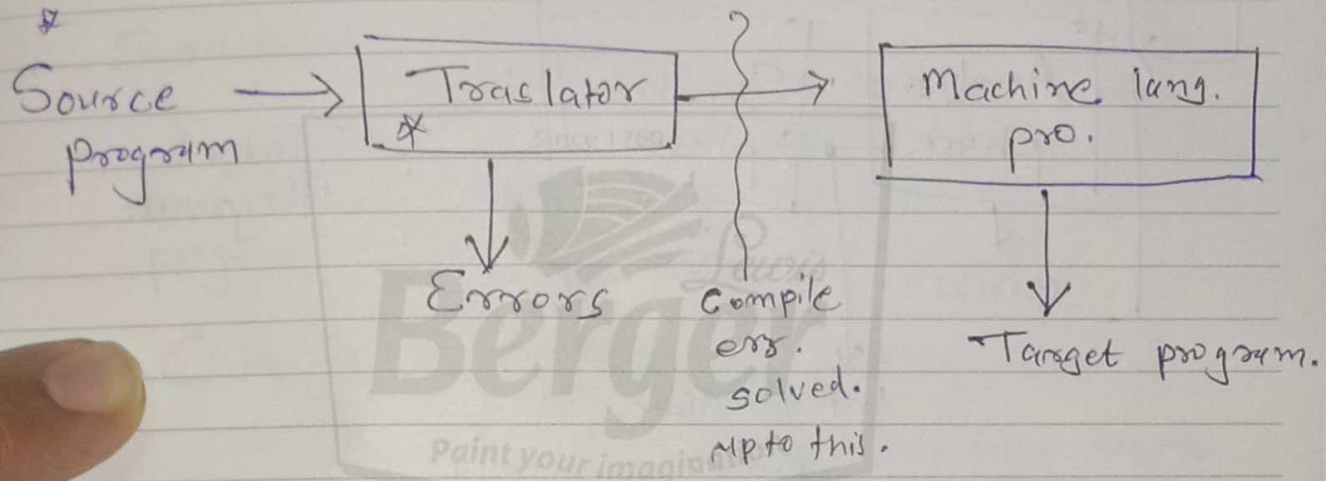
WEDNESDAY 01 MAY 2013

1. f1 | f2 | f3 | file ← relative path
2. ./file ← absolute path

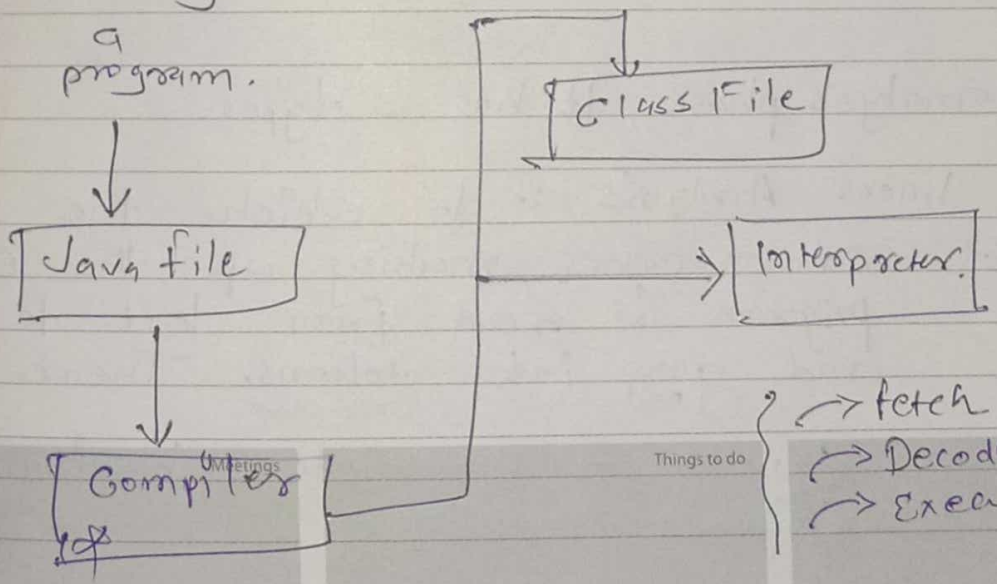
* Coding <script> </script> should write in <head></head>
* Js file as an External file.

LT [N.V.] * Program Translator.

*



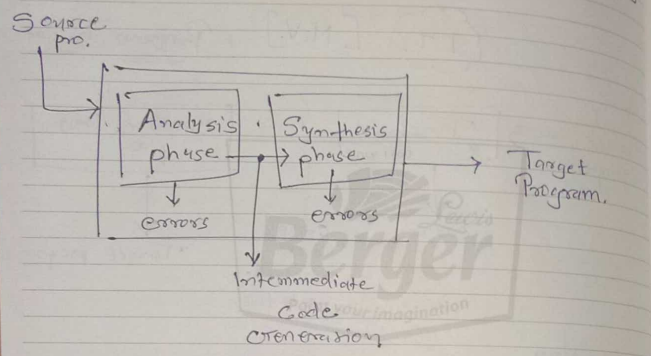

* writing a program.




Things to do | Meetings | important Calls

THURSDAY 02 2013 MAY

* lang. processing = [Analysis of Source prog.] + [Synthesis of target prg.]




⇒ In analysis phase, it has 3 types...

1. linear Analysis :- In which the stream of characters making up the source program is read from left to right and group into tokens. These are sequence of characters having a collective meaning.

- Linear analysis is also known as "lexical analysis."

FRIDAY 03 2013 MAY

2. Heirarchical Analysis :-

- In which character or tokens are group heirarchical into nested collections with collective meaning.
- Also known as "syntax analysis"

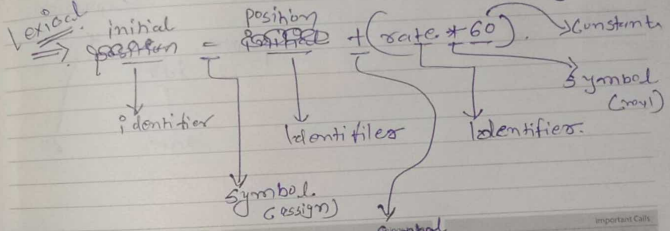
3. Symmentic Analysis :-

- In which certain checks are perform to ensure that the components of a program fit together meaningful.

(In this analysis, we check type).

Ex: Variable type checking.

Lexical:



SATURDAY

04

2013

MAY

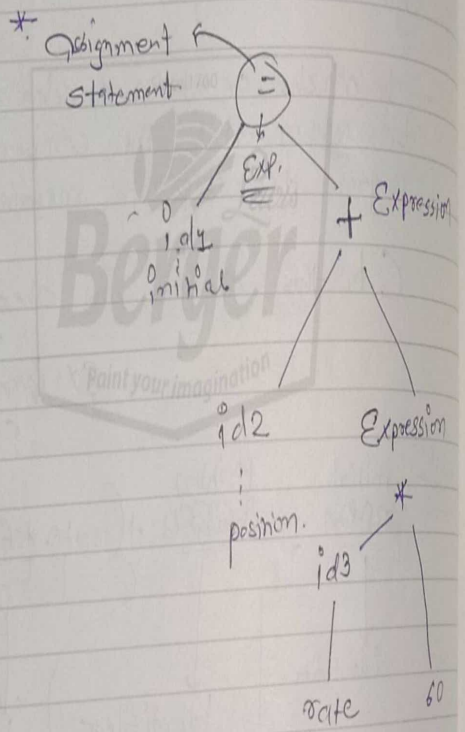
SUNDAY

05

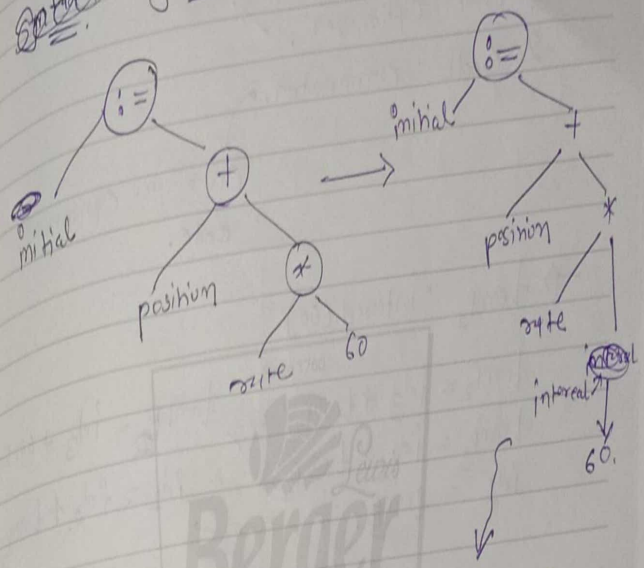
2013

MAY

Heirarchical Analysis:-




Syntax Symmetric:-




temp1 := inttoreal(60)
temp2 := rate * temp1
temp3 := position + temp2
initial := temp3.

Meetings

Things to do

Important Calls

MONDAY

06

2013
MAY

* Synthesis Analysis :- → to optimize the code
1. Code Optimizer.
2. Code Generator. → Generates Optimized code.

→ $temp_1$ = inttoreal(60)

$temp_2 = id_3 * temp_1 \Rightarrow temp_2 = id_3 * 60.0$
$temp_3 = id_2 * temp_2 \quad id_1 = id_2 + temp_3$
$id_1 = temp_3$

Convert into Assembly.

| | |
|---|---|
| MOVF $R_1$, $id_3$ | // $R_1 \leftarrow id_3$ |
| MULF $R_1$, #60.0 | // $R_1 = id_3 * 60.0$ |
| MOVF $R_2$, $id_2$ | // $R_2 \leftarrow id_2$ |
| ADDF $R_1$, $R_2$ | // $R_1 \leftarrow R_1 + R_2$ |
| MOVF $id_1$, $R_1$ | // $id_1 \leftarrow R_1$ |

TUESDAY

07

2013
MAY

* AOS [ANY]

*

Stack → 1. local/temp. variables
2. return address.
3. method fun parameters.

Heap → dynamically alloted.

Uninitialize / Initialize → global variables.

Text section → compile prog.
- prog. counter
- (remaining things).

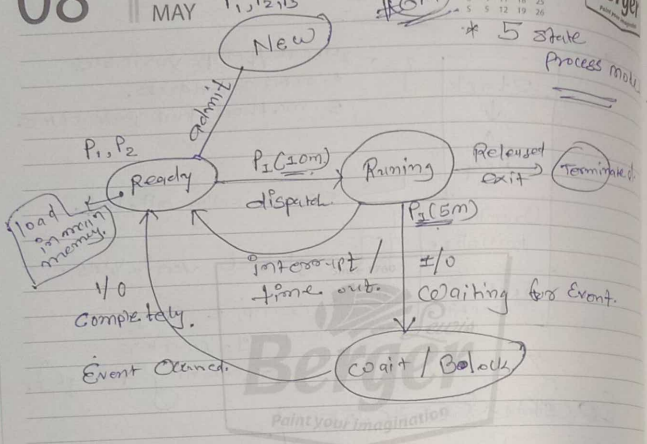
[OS Start a Process Model].

* Process ⇒ Active (main memory)
↓
program in Execution

* Program ⇒ Passive (Secondary mem.).

WEDNESDAY 08 2013 MAY

$P_1, P_2, P_3$ :- Processes...




* 5 state Process model.

* Need :- to make request for process(s).

* Ready :- - processes are ready to run.
- should go in to the main mem.
- In this, state multiple processes are possible.
- If in this state, any process is discarded then it is also ready state process.
- O.S. accepted that requests of process.

Meetings | Things to do | Important Calls

THURSDAY 09 MAY 2013

Hardik Pun... SE [Others Faculty]

* Fitness Application Requires :-

* Features :-
- scheduling
- git
- sample videos
- premium plan
- diff. sections according to Exercises.
- mentor's profile.
- User profile.
- Online Payment.
- Stopwatch.
- How much Cal. decrease
- Food guides.
- Suitable algorithm etc.

{superset platform} for placement.

Meetings | Things to do | Important Calls