COMPUTER ENGINEERING DEPARTMENT

ASSIGNMENT NO-01

SUB: SPCC

COURSE: T.E. **Year: 2020-2021** **Semester: VI**
DEPT: Computer Engineering
SUBJECT CODE: CSC602 **SUBMISSION DATE: 10/04/2021**

==========

Name: Amey Thakur **Roll No.: 50**

Class: TE Comps B-50 **ID: TU3F1819127**

Assignment No 1

Sr. No.	Question
1	Differentiate between the application program and system program. Indicate the order in which the following system programs are used from developing program up to their execution. Assembler, loader, linker, macro processor, compiler, editor.
2	What is System Programming? List some system programs and write their functions.
3	Explain the different error recovery techniques?
4	For the given grammar construct the operator precedence relation matrix E E+T/T T T*V/V V a/b/c/d Apply operator precedence parsing algorithm for statement a+b*c*d.
5	Explain the Different Code Optimization techniques for the compiler?
6	Draw and explain DAG and represent the following example with it. (a / b) + (a / b) * (c * d)

AMEY B 50 Amey

Page No.: Date: YOUVA

Q1 Differentiate between application program and system program. Indicate the order in which the following system programs are used from developing program upto its execution : Assembler, loader, linker, macro processor, compiler, editor.

Ans:

System Program	Application Program
① System software is any computer software which manages and controls computer hardware so that application software can perform a task.	① Application software is a program that enables end user to perform specific, productive tasks, such as word processing or image manipulation
② The function of system software is to manage the resources of the computer, automate its operation and facilitate program development. It is generally provided by the computer manufacturer or a specialized programming field.	② Application softwares are designed to perform specific data processing or computational tasks for the user. These programs are specifically designed to meet end user requirements.
③ System software is operating system which handles drivers	③ Application softwares are third party softwares that seek help from system software.
④ System software or operating system is the software that allows the computer to boot.	④ Application software are programs that run inside of or on top of the OS and allows you to do things. like email or word processing.

AMEY B 50 Amey

Page No.:	
Date:	YOUVA

⑤ System software performs tasks like transferring data from memory to disk or rendering text on to a display device	⑤ Application softwares are used to meet the user specific needs.
⑥ Ex: Operating systems loading programs, device drivers, compilers, assemblers & linkers and utility software	⑥ Ex: MS word, Oracle, Paint, Spreadsheets, etc.

Following is the order in which system programs are used from developing programs upto its execution.

① Editor

② Macro Processor
- It is a processor / program that reads a file and scans it for certain keywords. It is replaced by same text.

③ Compiler
- It is a computer program that transforms source code written in a programming language into another programming language.

④ Assembler
- It is a type of computer program that takes one or more object files and combines them into a single executable file that interprets software programs written in assembly language to machine language code and instructions that can be executed by computer.

⑤ Linker
- It is a computer program that takes one or more object files and combines them into single executable file.

AMEY B 50 Amey

Page No.: Youva
Date:

⑥ Loader

- Loader is a system program which takes object code as input and prepares it for execution.

System Program ⟶ Editor
Macro Processor
Compiler
Assembler
Linker
Loader

Target Program. ⟶ Executable File

AMEY B 50 Amey

Page No.: YOUVA
Date:

Q.2 What is system Programming ? List some system programs and write their functions.

Ans:

System Programming

- It is the activity of programming system software.
- It involves designing and writing computer programs that allow computer hardware to interface with the programmer and the user, leading to the effective execution of application software on the computer system.

Following are some system programs / softwares :

① Assemblers

- An assembler creates object code by translating assembly instructions into op-codes and by resolving symbolic names for memory locations and other entities.
- Ex : SPARC ASSEMBLERS, MASM ASSEMBLERS, MIPS ASSEMBLER

② Loaders

- Loading of the program in the memory for the purpose of execution is done by the loader.
- It is a program that accepts the object program and prepares these programs for execution by loading them into the memory

③ Linker

- A linker or a link editor is a program that takes one or more objects generated by a compiler or assembler and combines them into a single executable program.
- Relocation and linking of all modules is done by the linker.

AMEY B 50 Amey

Page No.:
Date:
YOUVA

④ Compilers

- It is a computer program that transforms source code written in a computer language (the source language) into another language (target language, often having a binary form known as object code.)
- The most common reason for having to transform source code is to create an executable program.

AMEY B 50 Amey

Page No.: YOUVA
Date:

Q.3. Explain the different error recovery techniques.

Ans:

Error Recovery Techniques:

① Panic Mode Recovery

- In this method, successive characters from an input are removed one at a time until a designated set of synchronizing tokens is found. Synchronizing tokens are delimeters such as ; or }.
- Advantage is that it is easy to implement and guarantees not to go in an infinite loop.
- Disadvantage is that considerable amount of time input is skipped without checking for additional errors.

② [Statement Mode Recovery] Phase Mode

- On error discovery of an error a parser may perform local correction on the remaining input. i.e. May replace prefix of the remaining input by some string that allows parser to continue
- Delete extra semicolon or insert missing.
- Its major drawback is in difficulty in copying with situation in which actual error is occured before the time of detection.

③ Error Production :

- Add rules to the grammar that describes the errorless syntax.

AMEY B 150 Amey

Page No.:
Date:
YOUVA

④ Global Correction:

- Compiler makes a few changes as possible in processing and in correct input string. There are algorithm for choosing a minimal sequence of changes to obtain globally least cost correction.
- Ex.: Given an incorrect input string x and grammar G, This algorithm will find the parse tree for a related string y such that the number of insertion, deletion and changes of tokens required to transform x → y is as small as possible.

AMEY B 50 Amey

Page No.: Date: YOUVA

Q.4. For the given grammar construct the operator precedence relation matrix

E E+T/T

T T*V/V

V a/b/c/d

Apply operator precedence parsing algorithm for statement a+b*c*d.

Ans:

(1) Given grammar is operator precedence grammar as it has the following properties.

- No RHS of any production has a ∈.
- No non-terminals are adjacent.

(2) Operator Precedence relation Table

	+	*	a	b	c	d	$
+	>	<	<	<	<	<	>
*	>	>	<	<	<	<	>
a	>	>	x	x	x	x	>
b	>	>	x	x	x	x	>
c	>	>	x	x	x	x	>
d	>	>	x	x	x	x	>
$	<	<	<	<	<	<	Accept

AMEY B 50 Amey

Page No.: Date: YOUVA

Stack	Input	Stack Top	Current Input	Action
$	a+b*c*d$	$	a	Shift a
$a	+b*c*d$	A	+	Reduce using V→a
$V	+b*c*d$	V	+	Reduce using T→V
$T	+b*c*d$	T	+	Reduce using E→T
$E	+b*c*d$	E	+	Shift +
$E+	b*c*d$	+	b	shift b
$E+b	*c*d$	B	*	Reduce using V→b
$E+V	*c*d$	V	*	Reduce using T→V
$E+T	*c*d$	T	*	shift *
$E+T*	c*d$	*	c	Shift c
$E+T*c	*d$	C	*	Reduce using V→c
$E+T*V	*d$	V	*	Reduce using T→T*V
$E+T	*d$	T	*	Shift *
$E+T*	d$	*	d	Shift d
$E+T*d	$	D	$	Reduce using V→d
$E+T*V	$	V	$	Reduce using T→T*V
$E+T	$	T	$	Reduce using E→E+T
$E	$	E	$	Accept

AMEY B 50 Amey:

Page No.:
Date:
YOUVA

Q.5 Explain the different code optimization techniques for the compiler.

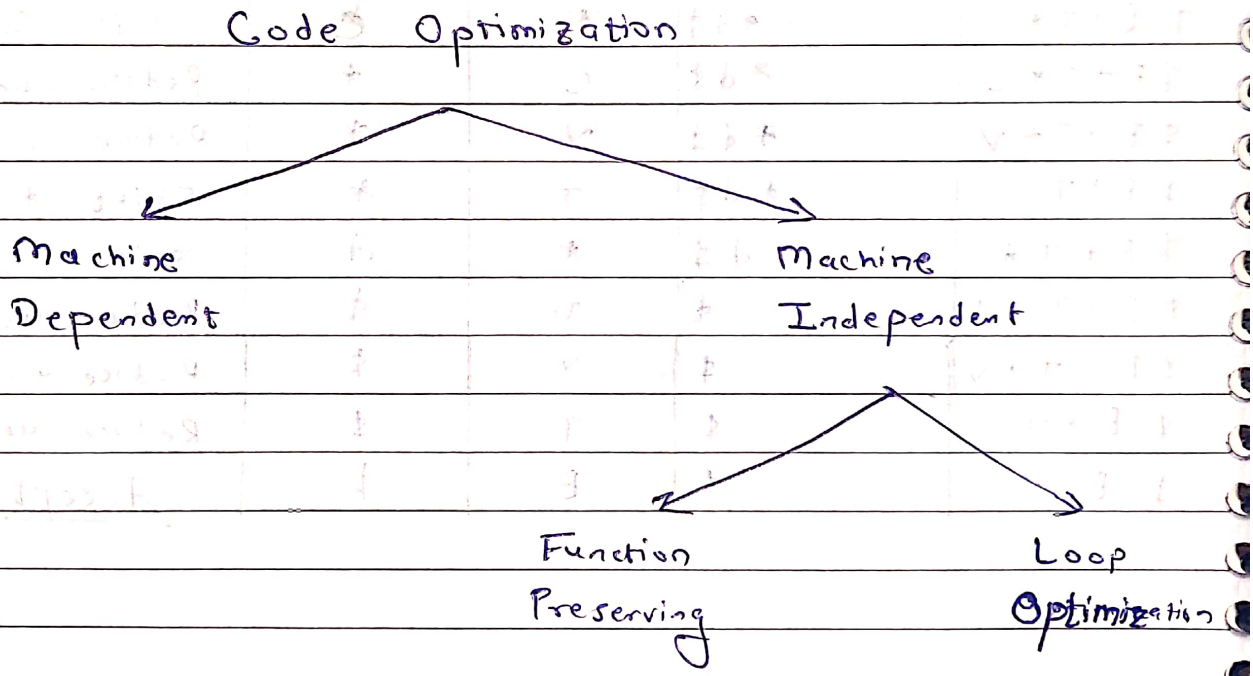
Ans:

Code Optimization

- Optimization is the procedure of converting a piece of code into more efficient code in terms of time and space complexity.

Optimization can be categorized into two types:

1. Machine Dependent
2. Machine Independent



AMEY B 50 Amey

Page No.:
Date:
YOUVA

Machine Dependent Optimization

- Machine dependent optimization is done after the target code has generated and when the code is transformed according to the target machine architecture. It improves memory performance machine dependent optimizers put efforts to take maximum advantage of memory hierarchy

Machine Independent Optimization

- There are two types of machine independent techniques.
  1. Function Preserving
  2. Loop optimization

Function Preserving

1. Common subexpression elimination
2. Constant Floding
3. Copy Propogation
4. Dead Code Elimination

Loop Optimization

1. Code motion
2. Strength reduction
3. Frequency reduction
4. Loop Distribution

AMEY B 50 Amey

Page No.:
Date:
YOUVA

Branch Optimization
- Rearranges the program code to minimize branching logic and to combine physically separate blocks of code.

Code Motion
- If variable used in a comprtation within a loop are not altered within the loop, the calculation can be performed outside of the loop and the results used within the loop.

Common subexpression elimination
- In common expressions the same value is recalculated in a subsequent expression. The duplicate expression can be eliminated by using a previous value.

Constant Propagation
- Constants used in an expression are combined and new ones are generated. Some implicit conversions between integers and floating point types are done.

Dead Code Elimination
- Eliminates code that cannot be reached or where the results are not subsequently used

Dead Store Elimination
- Eliminates store when the value stored is never referenced again. For ex, if two stores to the same location have no intervening load, the first store is unnecessary and is removed.

AMEY B. 50 Amey

Page No.: Date: YOUVA

Global Register Allocation
- Allocates variables and expressions to available hardware registers using a "graph coloring" algorithm.

Inlining
- Replaces function calls with actual program code.

Instruction Scheduling
- Reorders instructions to minimize execution time.

Interprocedural Analysis
- Uncovers relationships across function calls, and eliminates loads, stores and computations that cannot be eliminated with more straightforward optimizations.

Invariant IF code floating (Unswitching)
- Removes invariant branching code from loops to make more opportunity for other optimizations.

Reassociation
- Rearranges the sequence of calculations in an array subscript expression, producing more candidates. for common expression elimination.

Store Motion
- Moves store instruction out of loops.

Strength Reduction
- Replaces less efficient instructions with more efficient ones. For example, in array subscripting, an add instruction replaces a multiply instruction.

AMEY B 50 Amey

Page No.:
Date:
YOUVA

Value numbering

- It involves constant propagation, expression elimination and folding of several instructions into a single instruction.

AMFY B 50 Amey

Page No.: Date: YOUVA

Q.6 Draw and explain DAG and represent the following example with it.
$(a/b) + (a/b) * ((+d)$

Ans:

DAG

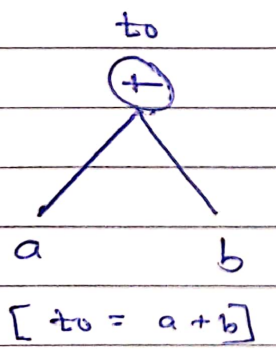
- Directed Acyclic Graph is a tool that depicts the structure of basic blocks. It helps to see the flow of values following among the basic blocks and offers optimization too.
- DAG provides easy transformation on basic blocks.
- DAG can be understood as:
  - → Leaf nodes represent identifiers, names or constants.
  - → Interior nodes represent the result of expression or the identifiers / name where the values are to be stored or assigned

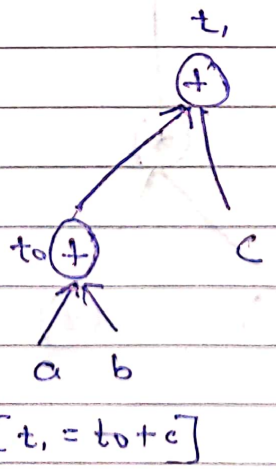
Example:

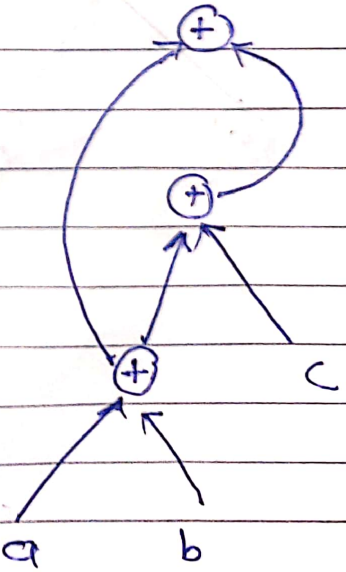
$t_0 = a + b$

$t_1 = t_0 + c$

$d = t_0 + t_1$



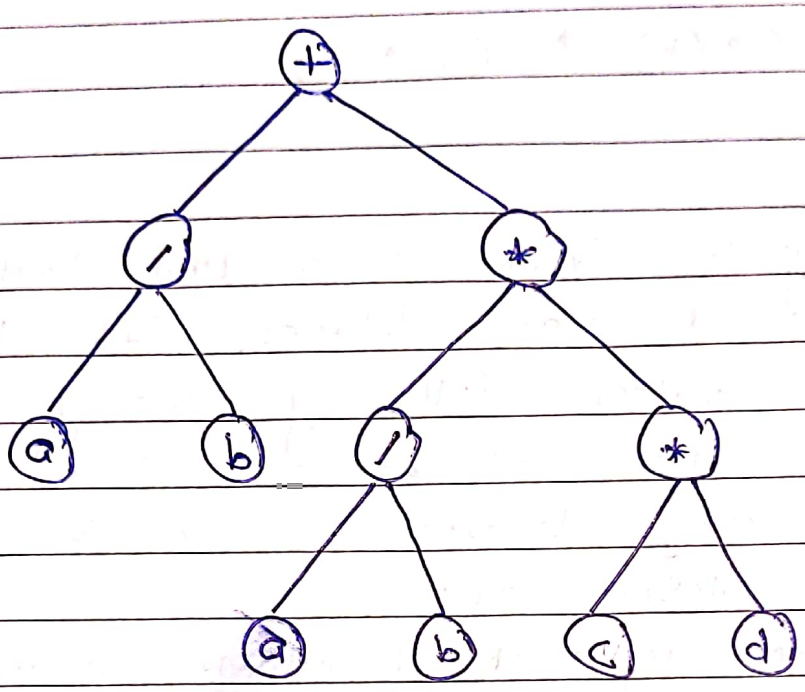




AMEY B 50 Amey

Page No.:
Date:

Example: (a/b) + (a * b) + c



Syntax Tree : (a/b) + ((a/b) + (c+d))