classmate 4.5 Bottom-Up parsing general style of bottom-up parsing its
shift-reduce parsing.
The process of bottom-up parsing is
to "reduce" a string we to the start symbol Sif is a grammer Consider the grunner: 5000 9/ p string agabab can be reduced to 5 by scarning for a substring that matches to the right side of some production Searing from left to right we get the following reduction stops. The above reduction is reverse of rightmost

Internally, a hardle of a string w is a substring that matches the right side of a production, and whose reduction to the non terminal on the left side of the production represent one step along the revere et a sightmost derivation. formally a handle of a right-sentential.

form & is a production A > B and a

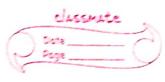
position of & whole the string B may be found and replaced by A to A produce the previous right-sentential from in a gightmost derivation of V.

Fire it 5 \* XAW => XBW

TM then A -> B in the position bellowing & is a handle of Xpw. Reducing B to A in & Bw can be thought of as "pruning the handle" Stack implementation of shift Reduce Parsing. - 2 data Aruchires used are. 1) Stack: It holds grannoe symbols. Initially it contains & only. 2) Input: It contains string to be parsed.

(ended with \$). Inhally When input buffer has with where When stack has \$5" and input has \$" parser announces successful parsing of string.

<u>-</u>	Till the input is either accorded
	Till the input is either accepted or error is seported, the parses Keeps taking the following 2 actions:
	He following 2 achieve
	Jan Jollowing Lawns
	9) shift: The next inset with
	a) Shift: The next input symbol is
	Shifted onto the Tos.
	he parter (on weating a hands
	on top of stack) replaces the
	b) keduce: The parter (on locating a hands. on top of stack) replaces the handle by the non terminal.
	Paris 1" 1 11" 1
	Parsing of "aabab" her grammes 5 + CC
	Coach
	15
	+
	aabab\$ shift
	\$a abab\$ Shift \$aa bab\$ Shift
	7
	A Comment of the comm
	1 reduce by Cos ac
	\$ C = ab \$ Sh.ft
	& Ca 64 SLAT.
	\$ Cab A Reduce by Cob
	7 760 / (390
	to by 5 occ
	\$ Accept.
	How are the alie of
	How are the choices of action (shift of
	Reduce are made so that the
	passes works correctly? There are 2 techniques:
	A) Operator Presed
-2.	A) Operator Precedence & B) LR Parsey



4.6 Operator - Precedence Parsing: The requirement for a grander to be on operator grander a are

i) It must be of from Type 2 of Chowky Hierarch (in CFG) ii) It cannot have E-production (of type A-re) iii) No production has 2 adjacent Variables/ Non-terminals. Mence 5 -> CC ) is not as E > E+E | E & E | (E) | id } is an operator grammy Operator precedence parses takes action based on manually created precedence relation. Relation Symbol Meaning a yields precedence to b a <. b [Achien - Shift] a " has same precedence as b [Achin - Shift] a "takes precedence over" 5 a > 6 [this - Reduce] The intention of precedence relations is to deliver the handle of a right-sentential form with i marking left-end = in the intercor and > marking right-end.



	The 2 ends of the string are
	The 2 ends of the string are marked by \$ with precedence
	'S Li every terminal of grammer
	and
	every terninal of grannal > \$
	Example: Consider grannos
	- The state of the
~~~	$E \rightarrow E + E \mid E \times E \mid id$
	Advaning
	i) & has higher precedence than 't'  ii) Both & and t' are left
	associatie.
	the precedence relation will be
<u></u> -	Incoming
	Top of terminal id + * \$  id > > >  id > > >
	stack terminal
	+ < > < >
	* < > > >
	\$ < < < /
	01 >1 1 1 0
- 	Placing solohin precedence relation for string id tid & determine
	handle.
	\$ < id > + < id > * < id > \$
- 	
4	



Reducing by E - id gives E+E\*E flacing precedence relation (by ignoring the Variables form terminals) まく・ナ く \* > \* Hence reduce by E + E \* E we get \$ < + > \$ Re Mence Sinelly reduce by E = E + E +0 with \$ E in stack & and "\$" in input the parser accepts the string as valid. Working of operator Precedence Parrage. 1) Initially top of stack contains & and input pointes points to the first symbol of w\$; where w is the imput 2) Let a' be the topmost terminal of stack at and 'b' be the input terminal. • If a < · b or a = 6 then a) push b' on top of stack and 6) Idvance to next input terminal. · It a > b then a) Repeatedly pop all terminals and non-tremidale a for top of stack until too terminal is to

	until the tos is a tesminal
	having le precedence that the
· · · · · · · · · · · · · · · · · · ·	last soped symbol.
	last poped symbol.  [ total & optionally consider the
	left variable of the production
	stack (Hearehealy)]
~	stack (Hearthealy)]
.,	3) Repeat step 2 till tos and input centain \$'.
	Cantaio \$
	Tasks
	1) Solve Q5 of Assignment 3
	The state of the s
	2) Refer to the Operator precedence
	relation der grammar
•	relation for grammar $E \to E + E / E + E / E - E / E / E / E / E / E / E / E /$
\	from book
- La Lora e 1	and hence parse if string
4	and hence parse ilp string id * (id + id) 1 id 1 id.
	3) Try to describe precedence pelahion for grammar.
	yammar.
	C> C + + c c !: c (c)   b
	where I de has higher precedence over!
	where I has higher precedence over!!  and both aperators are left association
1	V. T.
	The state of the s