		Ajm: Objective: About FIFo:
The EIFO algorithm is one of the simplest page replacement algorithms. It maintains a queue of pages in the order that were added to the memory, when a page needs to be replaced, the oldest page (the one at the frant of the queue) is removed	move from stage (HFO) Page	and implement the ge replacement As and implement Algorithm and it assed on page fau ting systems (as)

(1)

ST. VINCENT PALLOTTI COLLEGE OF ENGINEERING & TECHNOLOGY, NAGPUR - 441 108

key features: i) Simple to implement ii) Uses a queue to track the order cat page entries. iii) May suffer from Belady's anomaly", where increasing the number of frames can lead to more page faults. Important Terms: i) Page fault: Occurs when a page is already present in a frame. ii) Page thi: Occurs when the page is already present in a frame. iii) Trames: Slots in main memory to hold pages. Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: If the page is in the queue (Page hit) do nothing If the page is NOT in the queue (Page fault): The there is space in the queue add the lage. If the queue is factull, remove the front Page. Iii) Count the number of page faults		
entries. iii) May suffer from Belady's anomaly", where increasing the number of frames can lead to more page faults. Important Items: i) Page fault: Occurs when a page is already present in a frame. iii) Promes: Slots in main memory to hold pages. Algorithm: i) Tailfalize an empty queue for page frames ii) For each page is in the reference string: If the page is in the queue (Page hit), do nothing. If the page is NOT in the queue (Page fault): If there is space in the queue, add the lage.	Key features	Simple to implement
iii) May suffer from Belady's anomaly", where increasing the number of frames an lead to more page faults. Important Terms: i) Page fault: Occurs when a page is not found in the frame ii) Page tht: Occurs when the page is already present in a frame. iii) Frames: Slots in main memory to hold pages. Algorithm: i) Initialize an empty queue for page frames iii) For each page in the reference string: If the page is in the queue (Page hit), do nothing If there is space in the queue, add the lage. The the queue is feafull, remove the front Page. If the queue is feafull, remove the front Page.	The relies	11) Uses a queue to track the order cot agree
increasing the number of frames can head to more page faults. Important Terms: i) Page fault: Occurs when a page is not found in the frame ii) Page thit: Occurs when the page is already present in a frame. iii) Frames: Slots in main memory to hold pages. Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: If the page is in the queue (Page hit), do nothing To the page is NOT in the queue (Page fault). The page is space in the queue, add the lage. The queue is factull, remove the front Page.		374.63
Important Terms: is lage fault: Occurs when a page is not found in the frame in the frame iis lage thit: Occurs when the page is already present in a frame. iii) Frames: Slots in main memory to hold pages. Algorithm: is Initialize an empty queue for page frames ii) For each page in the reference string: If the page is in the queue (Page hit) do nothing If the page is NOT in the queue (Page fault): If there is space in the queue, add the lage lage If the queue is factull, remove the front lage It the queue is factull, remove the front lage	. Victory	iii) May suffer from Belady's gromaly" where
Important Terms: i) Page fault: Occurs when a page is not found in the frame in the frame ii) Page thit: Occurs when the page is already present in a frame. iii) Trames: Slots in main memory to hold pages. Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue (Page hit), do nothing — If there is space in the queue add the lage — If the queue is feefull, remove the front lage & insert the new page.		mercyany the number of tramps the last to
Important Terms: i) Page fault: Occurs when a page is not found in the frame in the frame ii) Page tht: Occurs when the page is already present in a frame. Iii) Frames: Slots in main memory to hold pages. Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue Page hit do nothing — If the page is NOT in the queue Page fault: — If there is space in the queue add the lage — The queue is factull, remove the front Page & insect the new page.		more page faults
Terms: 1) Page fault: Occurs when a page is not found in the frame ii) Page Hit: Occurs when the page is already present in a frame. iii) Frames: Slots in main memory to hold pages. Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue (Page hit), do nothing. — If the page is NOT in the queue (Page fault): — If there is space in the queue add the lage. Rage. Algorithm: I) Page is NOT in the queue (Page fault): — The page is not in the queue add the lage. Algorithm: I) Page is NOT in the queue add the lage.		
ii) Page Hit: Drives when the page is already present in a trame. iii) Frames: Slots in main memory to hold pages. Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue (Page hit), do nothing — If the page is NOT in the queue (Page fault): — If there is space in the queue, add the lage The the queue is fafull, remove the front Page & insert the new page.		
ii) Page Hit: Drives when the page is already present in a trame. iii) Frames: Slots in main memory to hold pages. Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue (Page hit), do nothing — If the page is NOT in the queue (Page fault): — If there is space in the queue, add the lage The the queue is fafull, remove the front Page & insert the new page.	Terms:	1) Page fault: Orius when 9 page is not found
Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue (Page hit) do nothing. — If the page is NOT in the queue (Page fault): — If there is space in the queue add the lage — The queue is feefull, remove the front lage & insert the new page.	2. 1	in the frame
Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue (Page hit) do nothing. — If the page is NOT in the queue (Page fault): — If there is space in the queue add the lage — The queue is feefull, remove the front lage & insert the new page.		11) Page Hit: Occurs when the page is already
Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue (Page hit) do nothing. — If the page is NOT in the queue (Page fault): — If there is space in the queue add the lage — The queue is feefull, remove the front lage & insert the new page.		present in a frame.
Algorithm: i) Initialize an empty queue for page frames ii) For each page in the reference string: — If the page is in the queue (Page hit) do nothing. — If the page is NOT in the queue (Page fault): — If there is space in the queue add the lage — The queue is feefull, remove the front lage & insert the new page.	ATT.	111) Frames : Slots in main memory to hold pages.
nothing. The page is NOT in the queue (Page fault): If there is space in the queue add the Rage The the queue is faifull, remove the front Page & insert the new page.		
nothing. The page is Not in the queue (Page fault): If there is space in the queue add the Rage The the queue is farfull, remove the front Page & insert the new page.	Algorithm:	i) Initialize an empty queue for page frames
nothing. The page is NOT in the queue (Page fault): If there is space in the queue add the Rage The the queue is faifull, remove the front Page & insert the new page.		in For each page in the reference string:
nothing The page is NOT in the queue Page fault: The there is space in the queue add the Page The queue is faifull, remove the front Page & insert the new page.		- If the page is in the gurue (Page hit) do
Page. The queue is faifull, remove the front Page & insert the new page.		nothing
Page. The queue is faifull, remove the front Page & insert the new page.		-If the page is NOT in the gurue (Page fault):
Page. The queue is faifull, remove the front Page & insert the new page.		- If there is space in the gueue add the
2 Insert the new page.		
& insert the new page.		If the queue is fa full remove the front Page
iii) Count the number of page faults		& insert the Dew Page.
		(ii) Count the number of page faults
Miles I am a second of the sec		

ST. VINCENT PALLOTTI COLLEGE OF ENGINEERING & TECHNOLOGY, NAGPUR - 441 108

Code	
Impenentation	mi Hinclude (iastream)
September 1	#include <queue></queue>
	# include (unordered_set)
· · · · · · · · · · · · · · · · · · ·	
	using namespare std;
and the second	void fifolage Replacement (int pages [] into
TO THE	int capacity
Togens of the Land	Unordered_set \(\text{int} \) \(\text{S} \) =
And the same of the	queue (int) index Vaeue.
he may be to	
	int page faults = 0;
	for (int i = 0 ; i < n ; i++){
The state of the s	if (s. size () < capacity) {
TE IT	if (s. stretind (pages [i] == s.end())}
	3.insert (pages [i]);
7年4月1年	indextrueue push (pages [i]);
	- pagesfault ++;
	}
	}else {
West of the second	if (s.find (pages [i]) == s.end ()){
te adds	int val = index Queue. Front ();
	index cluence-pap ().
	s.erase (yal')
CA CAS AFA	Sinsert (pages [1]);
Yearn and a second	1 Indec 1117:
ST. VIN	CENT PALLOTTI COLLEGE OF ENGINEERING & TECHNOLOGY, NAGPUR - 441 108

							79	
	Conclusion:							(1)
Sittessial And Market States	Conclusion. The FIFTO page implacement also is the simple & easy to implement but may not always provide the best performance. This kins experiment was executed &				The principal of the pr			
	3	14		in day to a last	9		And Andrews	
ST. VINCENT PALLOTTI CÖLL	Conclusion: The FIFO P to implement performance Buttessfully		elution	int.	int main ()	cout <<		

	3		
Ganchusion:	83 0/P:		
Bign: The Fifth Page riplusement algo. Is the simple Beary to implement but may not always provide the best performance. Thus this excultion was implemented successfully. St. VINCENT PALLOTTI COLLEGE OF ENGINEERING & TECHNOLOGY, NAGPUR-441 108	4	int pages [] = {1,3,0,3,5,6}. int n = Sizent (pages) / size at (pages [o]) int tapailty = 3; Itho Page Replatement (pages, n, rapaulty); Thum 0;	pageFault ++; } cout << Total Page Faults; '' << pageFaults << endl; } endl;