

CENG 140 / SECTION 2

HOMEWORK 02 – REPORT

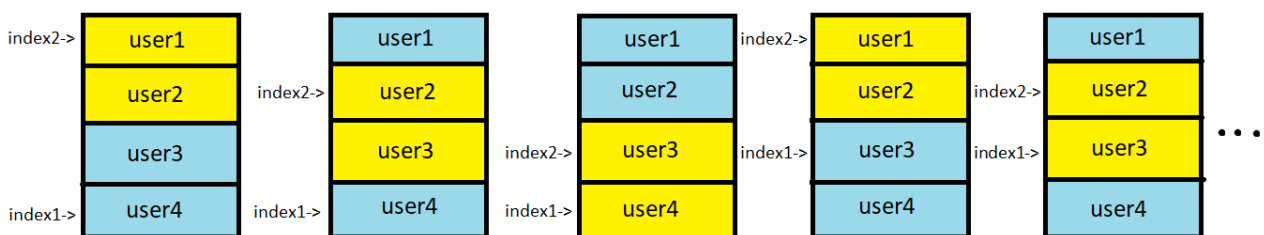
NAME: Çağdaş Fil

ID: 2093839

Part Two – Sorting and Searching Basics

1. sortBook

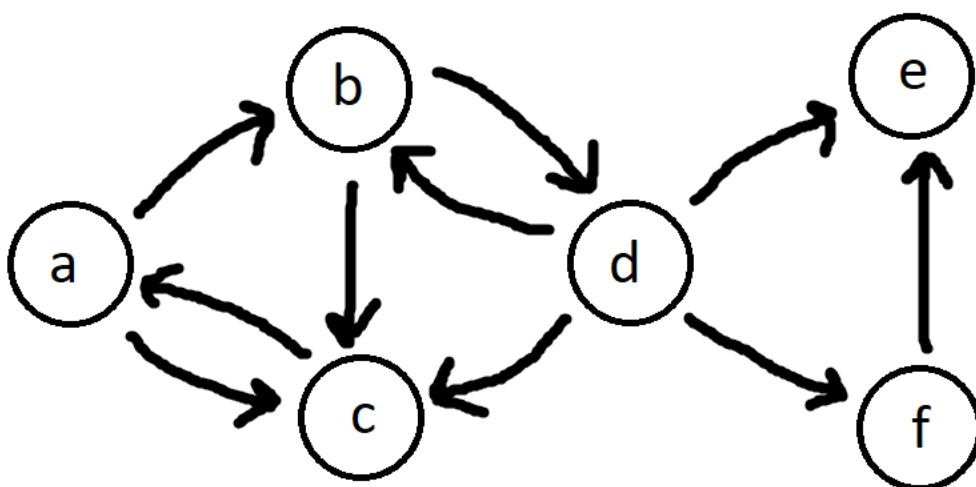
The sortBook function is used to sort the book according to the user's choice. The sortBook function takes three arguments, two pointer and a boolean value. The first pointer refers to a book, the second one refers to the comparator function which is chosen by user and the last argument is a boolean value that is either TRUE or FALSE. The sortbook function uses bubble sort algorithm to sort the book. Firstly, the function creates three integer namely index1, index2 and comparisonResult. index1 and index2 is used to search the book and comparisonResult is used to store the return value of the comparison function. The pointers userPtr1 and userPtr2 are used to point to choosen users in search operation. The index1 starts from the end, index2 starts from 0 to (index1-1) for every index1. The figure below shows how index1 and index2 work. The yellow boxes show the users that are choosen to compare. (The figure just shows how two users are choosen, possible swap operations are not shown.)



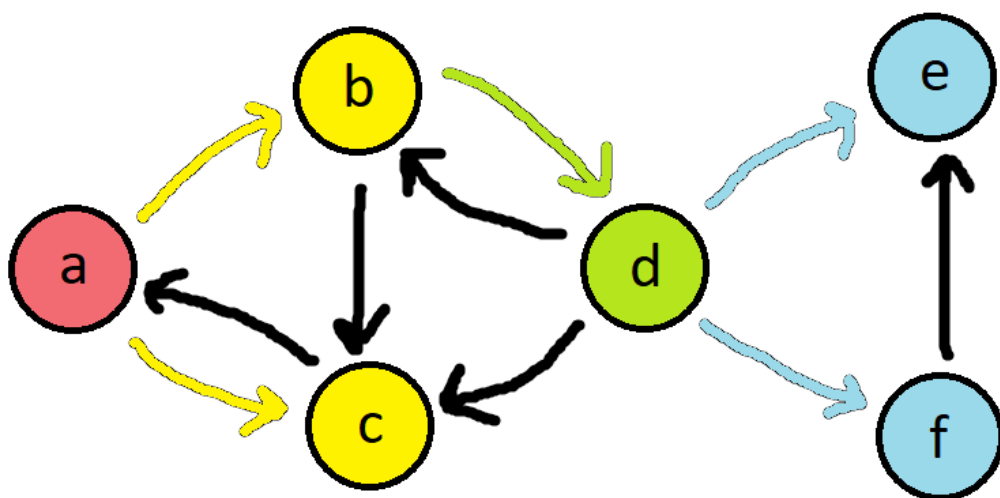
After two users are choosen, they are compared by comparison function. There are two comparison function, compareByName and compareByPopularity. The return value of these functions is in ascending order. Therefore, if the boolean value that sortBook takes is FALSE, the result is negated. Lastly, swap operation is performed if comparisonResult is positive.

2. shortestPath

The `shortestPath` function is used to find shortest path from a user to another user. The function takes three arguments, a pointer to the book, a pointer that refer to the start user, a pointer that refer to the target user. Briefly, the function starts from the start user and reaches the friends of the start user. Then, it reaches the friends of the friends of the start user and it goes like this. Until the target user is found, the function reaches the unreached users. Then, the users are being stored in the array, starting from the target to the start. A simple graph is shown below.



If we use `shortestPath` function to find shortest path from 'a' to 'e' in this graph:



Step by step computation will be like this:

a	a	a	a
	b	b	b
	c	c	c
		d	d
			e
			f

queue elements (step by step from left to right)

index (a)	-1	-1	-1	-1	-1
index (b)	-2	0	0	0	0
index (c)	-2	-2	0	0	0
index (d)	-2	-2	-2	1	1
index (e)	-2	-2	-2	-2	3
index (f)	-2	-2	-2	-2	-2

followerIndicesArray (step by step from left to right)

					<- return address
				address of user 'a'	
			address of user 'b'	address of user 'b'	
		address of user 'd'	address of user 'd'	address of user 'd'	
	address of user 'e'	address of user 'e'	address of user 'e'	address of user 'e'	
NULL	NULL	NULL	NULL	NULL	

friendPtrChainArray (step by step from left to right)

a