Assignment 12

Problem Statement. write a program to create a priority queue in ctt using morder-line to store the items in the queue. Create a class that includes the data items (which should be template) & the priority which should be into. The morder list should contain these objects, with x- operator overloaded so that the items with highest priority appear at the start of the list which will make it relatively only to retrive the highest Hem.)

1 objectives

or understand the priority queue data structure & its

implementation details.

7 Implement of operator overloading concept to compare two user defined data structures.

ordcornes

y Implement priority queue data structure using arrays data types

Hardware regularment:

Manufactures & Model: Acer 8018 3

Processor: Intel core 15 8th gen (82654 @ 1.66442)
Antalled memory: 8GB RAM, SIZ GB SSD

Architecturo: 64-bit

Reflecto seguizement:

operating system: Whinty 20.09 LTS on oracle virtual machine (3 processors & 4096MB base memory is allocated)

ctt version: ctt 14

compiler for ctt: 9th (version: 10.1.0) Code editor: sublime top (Build 2011)

Theory A priority queue is an abstract data type smilar to a regular queue / stack data structure in which each element additional has a "priority" associated with

In a priority queue, an element with high priority is served before an element with low priority.

There are there operations defined on priority queue.

zemove: remove the element with highfest priority

got front get the element with the highest privity which is present in the queue.

Bendo lode:

Operations of priority queue: Botes Algorithm front () of 11 returns element with highest retwee arrio]; perorety

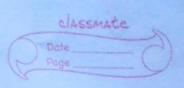
Algorithm mert (Tx) of 11 insert element in the queue if (lub is full) return:

1. add x to end of arr/list.

s. shift it to it's appropriate position known on

3. Increase the size of list

remove of 1/ removes highest priority Algorithm



if (but is empty) return. elsed 11 fort element of list is of lighest priority + shiet all deprode to their prestors forthow except 1st. I increase the value of front by ADT of classes: y class Job & // simulates job m operating system. private: string data; my priority: Job (string data = " 1, int priority = 0'& 1/constructor void seportar) & // sets value of data & priority String gelData() x the get Priority Of 11 return priority of current job Friend bool operator <= (Job, Job); 1/ overloaded <= operator to compare to jobs.

2) class friority Queue of 1/ template class of priority queue

private: Tarray to store elements of type T. int ins cubic: Priority Guerre () 1 // constructor function. into get size() & 11 return size of botal no. of elements in queue · // code bool is Empty () & 11 checks if queue is empty @ not 11 code 11 checks if capacity of que ue is allows any other new Hern bool isfull () & 1) wde 1/ return element of highest T Front UZ priority present in the queue 11 code void insect (Tx) of 11 insect x in the quelle. 11 tode void remove () / 1/ removes todelement of highest 11 code priority from queue. rord pemb () ? // paints all elements present in queue Heode

Analysis of algorithms:
operations of priority queue:

			CONTRACTOR OF THE PARTY OF THE
	O inserting.	in worst case our	contant space is
-	new element	time is required.	egused.
-	The same both	u is total elements	
2	A MONON TO DE	in queue.	
4	3 removing	time complexity of algorithm	Constant space is
1	element	is o(a) solver total	signised.
1	and represent that	descriptions there.	a shirt of
+	a acting highest	The algorithm zuns in	No entra space
+	3 getting highest priority element		11 organied.
+	1		

Applications The priorities queues are very helpful data stauture & trey have diverse applications. of Bandwidth monagement & Discrete event simulation

3> Dijkstras Algorithm - single source shortest path. 4) Huffman coding & frim's algorithm.

(onclyrions

data structure in many of algorithms of computer science.

The assignment gives complete I hight level idea of working of priority que ue data structure.