**DSA**

**LAB 3\_BT23CSE112 16/08/2024**

**Priority queue**

#include <stdio.h>

#include <stdlib.h>

struct node {

int data;

int priority;

struct node \*next;

};

struct node \*createnode() {

struct node \*temp = (struct node\*)malloc(sizeof(struct node));

printf("Enter the data : ");

scanf("%d", &temp->data);

printf("Enter the priority : ");

scanf("%d",&temp->priority);

temp->next = NULL;

return temp;

}

struct node\* createpq(int n) {

int i;

printf("Enter the elements :-\n");

struct node \*head = createnode();

struct node\* temp = head;

for (i = 1; i < n; i++) {

struct node\* newnode = createnode();

if (newnode->priority < head->priority) {

newnode->next = head;

head = newnode;

} else {

temp = head;

while (temp->next != NULL && newnode->priority > temp->next->priority) {

temp = temp->next;

}

newnode->next = temp->next;

temp->next = newnode;

}

}

return head;

}

void traverse(struct node\* start) {

struct node\* ptr = start;

printf("The elements of the priority queue are:\n");

while (ptr != NULL) {

printf("%d\t", ptr->data);

ptr = ptr->next;

}

printf("\n");

}

int pop(int n, struct node \*head) {

struct node \*temp = head;

for (int i = 0; i < n-1 && temp != NULL; i++) {

temp = temp->next;

}

if (temp == NULL) {

printf("Position out of range\n");

return -1;

} else {

return temp->data;

}

}

int main() {

int l, pos;

printf("Enter the length of queue: ");

scanf("%d", &l);

struct node\* head = createpq(l);

traverse(head);

printf("Enter the position that you want to see: ");

scanf("%d", &pos);

if (pos > l || pos < 1) {

printf("Error!! Enter the correct position\n");

} else {

printf("\n%d", pop(pos, head));

}

return 0;

}

