Q: IMPLEMENT PRIORITY QUEUE USING LINKED LISTS

CODE :

#include <stdio.h>

#include <stdlib.h>

struct node

{

    int data;

    int priority;

    struct node \*next;

    struct node \*prev;

} \*front, \*rear, \*temp;

void enqueue(int data, int priority);

void print();

int main()

{

    front = NULL;

    rear = NULL;

    temp = NULL;

    int n;

    printf("Enter the number of node you want to add in the priority queue :  ");

    scanf("%d", &n);

    while (n--)

    {

        int data;

        int priority;

        printf("Enter the the data and its priority :\n");

        printf("data -> : \n");

        scanf("%d",&data);

        printf("Priority -> : \n");

        scanf("%d",&priority);

        enqueue(data, priority);

    }

     print();

    return 0;

}

void enqueue(int data, int priority)

{

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

    ptr->data = data;

    ptr->priority = priority;

    ptr->next = NULL;

    ptr->prev = NULL;

    if (front == NULL && rear == NULL)

    {

        front = ptr;

        rear = ptr;

        temp = front;

    }

    else

    {

        if (ptr->priority > temp->priority)

        {

            while (  temp->next != NULL  && temp->next->priority < ptr->priority)

            {

                temp = temp->next;

            }

            if (temp->next == NULL)

            {

                temp->next = ptr;

                ptr->prev = temp;

                ptr->next = NULL;

                rear = ptr ;

            }

            else

            {

                ptr->prev = temp;

                ptr->next = temp->next;

                temp->next->prev = ptr;

                temp->next = ptr;

            }

        }

        else if (ptr->priority < temp->priority)

        {

            while (  temp->prev != NULL && temp->prev->priority > ptr->priority )

            {

                temp = temp->prev;

            }

            if (temp->prev == NULL)

            {

                ptr->next = temp;

                ptr->prev = NULL;

                temp->prev = ptr;

                front = ptr ;

            }

            else

            {

                ptr->next = temp;

                ptr->prev = temp->prev;

                temp->prev->next = ptr;

                temp->prev = ptr;

            }

        }

    }

}

void print()

{

struct node \* ptr = front ;

printf("The Entered values  in priority :");

while(ptr != NULL)

{

    printf("%d ",ptr->data);

    ptr = ptr->next;

}

}

OUTPUT :

Enter the number of node you want to add in the priority queue : 5

Enter the the data and its priority :

data -> :

764

Priority -> :

1

Enter the the data and its priority :

data -> :

564

Priority -> :

5

Enter the the data and its priority :

data -> :

45

Priority -> :

4

Enter the the data and its priority :

data -> :

78

Priority -> :

2

Enter the the data and its priority :

data -> :

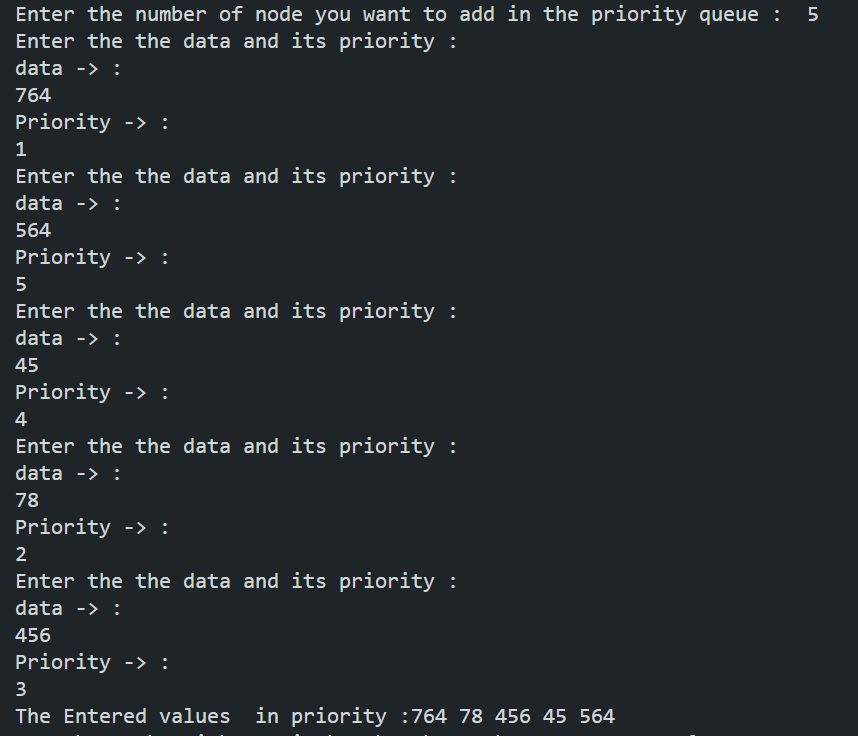
456

Priority -> :

3

The Entered values in priority :764 78 456 45 564

SCREENSHOT :



TIME COMPLEXITY :

O(n^2)

Due to while loop in main O(n) & Enqueue O(n) so final time complexity : O(n^2)