Name: Patil Krushna Gopal

Roll no.: 38

//Write a program for creating Max Heap using ADJUST/HEAPIFY

#include<iostream>

#include<conio.h>

using namespace std;

class AdjustMaxHeap

{

private:

    int a[10], n, i;

public:

    void Adjust(int a[], int i, int n);

    void Heapify(int a[], int n);

    void get();

    void show();

};

void AdjustMaxHeap::get()

{

    cout << "Enter the size of array : ";

    cin >> n;

    cout << "Enter " << n << " element : ";

    for (int b = 1; b <= n; b++)

    {

        cin >> a[b];

    }

    Heapify(a, n);

}

void AdjustMaxHeap::Heapify(int a[], int n)

{

    for (i = (n / 2); i >= 1; i--)

    {

        Adjust(a, i, n);

    }

}

void AdjustMaxHeap::Adjust(int a[], int i, int n)

{

    int j, item;

    j = 2 \* i;

    item = a[i];

    while (j <= n)

    {

        if (j < n && a[j] < a[j + 1])

        {

            j = j + 1;

        }

        if (item >= a[j])

        {

            return;

        }

        else

        {

            a[j / 2] = a[j];

            j = 2 \* j;

        }

    }

    a[j / 2] = item;

}

void AdjustMaxHeap::show()

{

    cout << "element after using adjust heapify : ";

    for (int c = 1; c <= n; c++)

    {

        cout << a[c] << "\t";

    }

}

int main()

{

    AdjustMaxHeap obj;

    obj.get();

    obj.show();

    return(0);

}

//Output:

Enter the size of array : 7

Enter 7 element : 40 80 35 90 45 50 70

element after using adjust heapify : 90 80 70 40 45 50 35

//Write a program for creating Min Heap using ADJUST/HEAPIFY

#include<iostream>

#include<conio.h>

using namespace std;

class AdjustMinHeap

{

private:

    int a[10], n, i;

public:

    void Adjust(int a[], int i, int n);

    void Heapify(int a[], int n);

    void get();

    void show();

};

void AdjustMinHeap::get()

{

    cout << "Enter the number of nodes : ";

    cin >> n;

    cout << "Enter " << n << " nodes : ";

    for (int b = 1; b <= n; b++)

    {

        cin >> a[b];

    }

    Heapify(a, n);

}

void AdjustMinHeap::Heapify(int a[], int n)

{

    for (i = (n / 2); i >= 1; i--)

    {

        Adjust(a, i, n);

    }

}

void AdjustMinHeap::Adjust(int a[], int i, int n)

{

    int j, item;

    j = 2 \* i;

    item = a[i];

    while (j <= n)

    {

        if (j < n && a[j] > a[j + 1])

        j++;

        if (item <= a[j])

        break;

        a[j / 2] = a[j];

        j = 2 \* j;

    }

    a[j / 2] = item;

}

void AdjustMinHeap::show()

{

    cout << "element after using adjust heapify : ";

    for (int c = 1; c <= n; c++)

    {

        cout << a[c] << "\t";

    }

}

int main()

{

    AdjustMinHeap obj;

    obj.get();

    obj.show();

    return 0;

}

Output :

Enter 7 nodes : 40 80 35 90 45 50 70

element after using adjust heapify : 35 45 40 90 80 50 70