

Roll No: 322065

Batch : B3

GR No : 21920048

Name : Vikram santosh shinde

Optimal Merge Pattern using Greedy approach

Program:

```
#include<bits/stdc++.h>
```

```
using namespace std;
```

```
void optimalCost(int size, int files[])
```

```
{
```

```
    priority_queue<int, vector<int>,
```

```
        greater<int>> pq;
```

```
    for(int i = 0; i < size; i++)
```

```
    {
```

```
        pq.push(files[i]);
```

```
    }
```

```
    int count = 0;
```

```

cout<<"Passwise output:"<<endl;
while(pq.size() > 1)
{

    int first_smallest = pq.top();
    pq.pop();
    int second_smallest = pq.top();
    pq.pop();

    int temp = first_smallest + second_smallest;

    count += temp;
    cout<<first_smallest<<"+"<<second_smallest<<"="<<temp<<endl;

    pq.push(temp);
}
cout<<"Minimum computations to be done : "<<count;
}

int main()
{

    int n,i;

    cout<<"Enter no of files : "<<endl;

```

```
        cin>>n;

int files[n];

        cout<<"Enter file sizes :"<<endl;

        for(i=0;i<n;i++){

            cin>>files[i];

        }


        optimalCost(n, files);


        return 0;

}
```

Output:

```
Enter no of files :
5
Enter file sizes :
10
20
30
5
30
Passwise output:
5+10=15
15+20=35
30+30=60
35+60=95
Minimum computations to be done :205

...Program finished with exit code 0
Press ENTER to exit console. █
```