

```
# include <iostream>

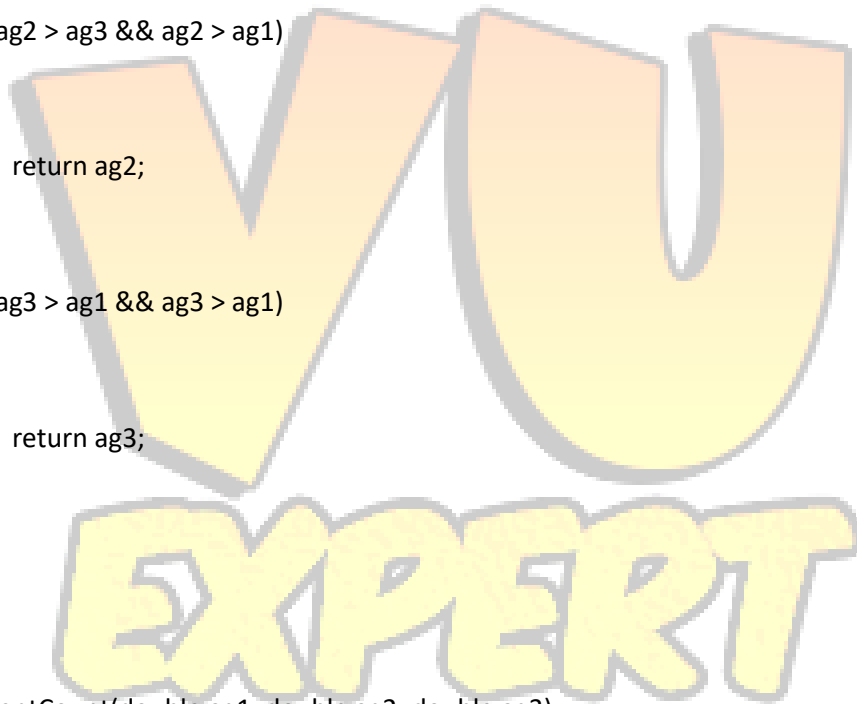
using namespace std;

double maxAggregate(double ag1, double ag2, double ag3)
```

```
{
    if(ag1 > ag2 && ag1 > ag3)
    {
        return ag1;
    }
    else if(ag2 > ag3 && ag2 > ag1)
    {
        return ag2;
    }
    else if(ag3 > ag1 && ag3 > ag2)
    {
        return ag3;
    }
}
```

```
int eligibleStudentCount(double ag1, double ag2, double ag3)
```

```
{
    double temp[3] = {ag1, ag2, ag3};
    int count = 0;
    for(int i = 0; i < 3; i++)
    {
```



```
        if(temp[i] >= 90)
        {
            count++;
        }
    }
    return count;
}
```

```
double aggSum(double ag1, double ag2, double ag3)
```

```
{
    double temp[3] = {ag1, ag2, ag3};
    double sum = 0;
    for(int i = 0; i < 3; i++)
    {
        sum += temp[i];
    }
    return sum;
}
```

```
main()
```

```
{
```

```
    double st1_SSCMarks = 971, st1_HSSCMarks = 953, st1_EntryTestMarks = 950;
```

```
    double st2_SSCMarks = 1052, st2_HSSCMarks = 976, st2_EntryTestMarks = 996;
```

```
double st3_SSCMarks = 990, st3_HSSCMarks = 981, st3_EntryTestMarks = 1020;
```

```
double st1_AggPerc, SSC_Agg_st1, HSSC_Agg_st1, Entry_Agg_st1, Aggregate_st1;
```

```
double st2_AggPerc, SSC_Agg_st2, HSSC_Agg_st2, Entry_Agg_st2, Aggregate_st2;
```

```
double st3_AggPerc, SSC_Agg_st3, HSSC_Agg_st3, Entry_Agg_st3, Aggregate_st3;
```

```
SSC_Agg_st1 = st1_SSCMarks * .1;
```

```
HSSC_Agg_st1 = st1_HSSCMarks * .4;
```

```
Entry_Agg_st1 = st1_EntryTestMarks * .5;
```

```
Aggregate_st1 = SSC_Agg_st1 + HSSC_Agg_st1 + Entry_Agg_st1;
```

```
st1_AggPerc = Aggregate_st1 / 1100 * 100;
```

```
SSC_Agg_st2 = st2_SSCMarks * .1;
```

```
HSSC_Agg_st2 = st2_HSSCMarks * .4;
```

```
Entry_Agg_st2 = st2_EntryTestMarks * .5;
```

```
Aggregate_st2 = SSC_Agg_st2 + HSSC_Agg_st2 + Entry_Agg_st2;
```

```
st2_AggPerc = Aggregate_st2 / 1100 * 100;
```

```
SSC_Agg_st3 = st3_SSCMarks * .1;
```

```
HSSC_Agg_st3 = st3_HSSCMarks * .4;
```

```
Entry_Agg_st3 = st3_EntryTestMarks * .5;
```

```
Aggregate_st3 = SSC_Agg_st3 + HSSC_Agg_st3 + Entry_Agg_st3;
```

```
st3_AggPerc = Aggregate_st3 / 1100 * 100;
```

```
        cout << "*****Merit
Calculation*****" << endl << endl;

        cout << "\n";

        cout << "\nPercentage of Student One Calculate as: " << st1_AggPerc << endl;

        cout << "Percentage of Student One Calculate as: " << st2_AggPerc << endl;

        cout << "Percentage of Student One Calculate as: " << st3_AggPerc << endl;

        cout << "Maximum aggregate percentage is: " << maxAggregate(st1_AggPerc, st2_AggPerc,
st3_AggPerc) << endl;

        cout << "Number of Students Eligible for Admission are: " << eligibleStudentCount(st1_AggPerc,
st2_AggPerc, st3_AggPerc) << endl;

        cout << "Aggregate sum of all students are: " << aggSum(st1_AggPerc, st2_AggPerc,
st3_AggPerc) << endl;
    }
}
```

