WEEK3

P1:

```
#include <iostream>
using namespace std;
inline int cube(int number) { return number * number * number; }
int main() {
int num;
cin >> num;
cout << cube(num) << endl;
return 0;
}
```

P2:

```
#include <iostream>
#include <cstring>
using namespace std;

bool isPalindrome(const char str[]) {
int length = strlen(str);
for (int i = 0; i < length / 2; ++i)
if (str[i] != str[length - 1 - i])
return false;
return true;
}

int main() {
char str[100];
cin >> str;
if (isPalindrome(str))
cout << "Palindrome" << endl;
else
```

```
cout << "Not a palindrome" << endl;
return 0;
}
```

P3:

```
#include <iostream>
using namespace std;

void swap(int &a,int &b){int temp=a;a=b;b=temp;}
void swap(float &a,float &b){float temp=a;a=b;b=temp;}
void swap(char &a,char &b){char temp=a;a=b;b=temp;}

int main(){
    int a,b;
    float x,y;
    char p,q;
    cout<<"Enter two integers";
    cin>>a>>b;
    cout<<"Enter two Float values";
    cin>>x>>y;
    cout<<"Enter two characters";
    cin>>p>>q;
    swap(a,b);
    swap(x,y);
    swap(p,q);

    cout<<"Swapped integers: "<<a<<" "<<b<<endl;
    cout<<"Swapped floats: "<<x<<" "<<y<<endl;
    cout<<"Swapped characters: "<<p<<" "<<q<<endl;

    return 0;
```

```
}
```

P4:

```
#include <iostream>
using namespace std;
int inverse(int num){
	int result=0;
	while(num>0){
		result=result*10+num%10;
		num/=10;
	}
	return result;
}
int doubleOfInverse(int num){
	return 2*num;
}
int main(){
	int num;
	cout<<"Enter an integer: ";
	cin>>num;
	int inv=inverse(num);
	int doubledInv=doubleOfInverse(inv);
	cout<<"Inverse: "<<inv<<endl;
	cout<<"Double of inverse: "<<doubledInv<<endl;
	return 0;
}
```

P5:

```
#include <iostream>
using namespace std;
struct EMPLOYEE{
```

```
    int Employee_Number;
    char Employee_Name[50];
    float Basic,DA,IT,Net_Sal;
};
void readData(EMPLOYEE &e){
    cout<<"Enter Employee Number: ";
    cin>>e.Employee_Number;
    cout<<"Enter Employee Name: ";
    cin.ignore();cin.getline(e.Employee_Name,50);
    cout<<"Enter Basic Salary: ";
    cin>>e.Basic;
}
void calculateNetSalary(EMPLOYEE &e){
    e.DA=0.12*e.Basic;
    float grossSalary=e.Basic+e.DA;
    e.IT=0.18*grossSalary;
    e.Net_Sal=grossSalary-e.IT;
}
void displayNetSalary(const EMPLOYEE &e){
    cout<<"Employee Number: "<<e.Employee_Number<<endl;
    cout<<"Employee Name: "<<e.Employee_Name<<endl;
    cout<<"Net Salary: "<<e.Net_Sal<<endl;
}
int main(){
    int N;
    cout<<"Enter number of employees: ";
    cin>>N;
    EMPLOYEE employees[N];
    for(int i=0;i<N;i++){
        cout<<"Enter details for employee "<<i+1<<endl;
        readData(employees[i]);
```

```
            calculateNetSalary(employees[i]);
        }
        for(int i=0;i<N;i++){
            cout<<"Details of employee "<<i+1<<endl;
            displayNetSalary(employees[i]);
        }
        return 0;
}
```

Bonus1:

```
#include <iostream>
#include <cmath>
using namespace std;
int main(){
int n;
cout<<"Enter number of data points: ";
cin>>n;
float data[n];
float sum=0;
cout<<"Enter the data points: ";
for(int i=0;i<n;++i){
cin>>data[i];
sum+=data[i];
}
float mean=sum/n;
float varianceSum=0;
for(int i=0;i<n;++i){
float diff=data[i]-mean;
varianceSum+=diff*diff;
}
float variance=varianceSum/n;
```

```
float stdDev=sqrt(variance);
cout<<"Mean: "<<mean<<endl;
cout<<"Variance: "<<variance<<endl;
cout<<"Standard Deviation: "<<stdDev<<endl;
return 0;
}
```

Bonus 2

```
#include <iostream>
#include <climits>
using namespace std;

int main(){
const int FLOORS=5;
const int ROOMS=8;
int occupancy[FLOORS][ROOMS];
int totalOccupants=0,totalAdults=0,totalChildren=0;
int totalFloorOccupancy[FLOORS]={0};
int totalFloorAdultOccupancy[FLOORS]={0};
int totalFloorChildOccupancy[FLOORS]={0};
for(int i=0;i<FLOORS;++i){
cout<<"Enter number of people in each room on floor "<<i+1<<":"<<endl;
for(int j=0;j<ROOMS;++j){
int numPeople;
cout<<"Room "<<j+1<<": ";
cin>>numPeople;
occupancy[i][j]=numPeople;
totalOccupants+=numPeople;
totalFloorOccupancy[i]+=numPeople;
}
}
```

```
for(int i=0;i<FLOORS;++i){
cout<<"Enter number of adults and children in each room on floor "<<i+1<<":"<<endl;
for(int j=0;j<ROOMS;++j){
int numAdults,numChildren;
cout<<"Room "<<j+1<<": ";
cout<<"Adults: ";
cin>>numAdults;
cout<<"Children: ";
cin>>numChildren;
totalAdults+=numAdults;
totalChildren+=numChildren;
totalFloorAdultOccupancy[i]+=numAdults;
totalFloorChildOccupancy[i]+=numChildren;
}
}
int maxOccupancy=INT_MIN,minOccupancy=INT_MAX;
int maxFloor=0,minFloor=0;
for(int i=0;i<FLOORS;++i){
if(totalFloorOccupancy[i]>maxOccupancy){
maxOccupancy=totalFloorOccupancy[i];
maxFloor=i+1;
}
if(totalFloorOccupancy[i]<minOccupancy){
minOccupancy=totalFloorOccupancy[i];
minFloor=i+1;
}
}
cout<<"Total Number of occupants = "<<totalOccupants<<endl;
cout<<"Total Number of adults = "<<totalAdults<<endl;
cout<<"Total Number of children = "<<totalChildren<<endl;
for(int i=0;i<FLOORS;++i){
```

```
cout<<"Total Floor "<<i+1<<" occupancy = "<<totalFloorOccupancy[i]<<endl;
cout<<"Total Floor "<<i+1<<" adult occupancy = "<<totalFloorAdultOccupancy[i]<<endl;
cout<<"Total Floor "<<i+1<<" child occupancy = "<<totalFloorChildOccupancy[i]<<endl;
}
cout<<"Max. Occupancy is on floor "<<maxFloor<<endl;
cout<<"Min. Occupancy is on floor "<<minFloor<<endl;
return 0;
}
```

Bonus 3:

```
#include <iostream>
#include <cmath>
using namespace std;
int main(){
int n;
cout<<"Enter the number of points: ";
cin>>n;
float x[n],y[n];
cout<<"Enter the points (x y):"<<endl;
for(int i=0;i<n;i++){
cout<<"Point "<<i+1<<": ";
cin>>x[i]>>y[i];
}
float xk,yk;
cout<<"Enter the coordinates of the input point (xk yk): ";
cin>>xk>>yk;
int nearestIndex=0;
float minDistance=pow(x[0]-xk,2)+pow(y[0]-yk,2);
for(int i=1;i<n;i++){
float distance=pow(x[i]-xk,2)+pow(y[i]-yk,2);
if(distance<minDistance){
```

```
minDistance=distance;
nearestIndex=i;
}
}
cout<<"The nearest point is ("<<x[nearestIndex]<<", "<<y[nearestIndex]<<")"<<endl;
return 0;
}
```