

```

#include<iostream>

using namespace std;

void pointerPassByValue(int * ptr){

    cout<<endl;
    cout<<endl;
    cout<<"POINTER PASS BY VALUE"<<endl;
    cout<<"Adress of ptr: "<<&ptr<<endl;
    cout<<"Address stored in ptr: "<<ptr<<endl;
    cout<<"Value pointer by ptr: "<<*ptr<<endl;

}

void pointerPassByReference(int * &ptr){

    cout<<"POINTER PASS BY REFERENCE"<<endl;
    cout<<"Adress of ptr: "<<&ptr<<endl;
    cout<<"Address stored in ptr: "<<ptr<<endl;
    cout<<"Value pointer by ptr: "<<*ptr<<endl;

}

void myFun(int &myVar){

    cout<<"address of myVar:      "<<&myVar<<endl;
    cout<<"value stored in myVar:   "<<myVar<<endl;
}

int main(){

/*

    int a = 3;

    cout<<"value stored in a:      "<<a<<endl;
    cout<<"address of a :          "<<&a<<endl;
    myFun(a);

    */

/*

void pointerPassByValue(int * ptr){
    cout<<endl;
    cout<<endl;
    cout<<"POINTER PASS BY VALUE"<<endl;
    cout<<"Adress of ptr: "<<&ptr<<endl;
    cout<<"Address stored in ptr: "<<ptr<<endl;
    cout<<"Value pointer by ptr: "<<*ptr<<endl;
}

```

```

*/
cout<<"POINTER IN MAIN FUNCTION"<<endl;

int *newptr = new int;
*newptr = 2939;

cout<<"Adress of ptr: "<<&newptr<<endl;
cout<<"Address stored in ptr: "<<newptr<<endl;
cout<<"Value pointer by ptr: "<<*newptr<<endl;
    pointerPassByValue(newptr);
/*
int x = 5;

cout<<"Value stored in x:"<<x<<endl;
cout<<"Address of x: "<<&x<<endl;

int *ptr = &x;

cout<<"Value pointed by ptr: "<<*ptr<<endl;
cout<<"Address stored in ptr:"<<ptr<<endl;
cout<<"Address of ptr:"<<&ptr<<endl;

int **ptr2 = &ptr;

cout<<"Value pointed by ptr2: "<<**ptr2<<endl;
cout<<"Address stored in ptr2: "<<ptr2<<endl;

cout<<"Address of ptr2: "<<&ptr2<<endl;

int *anotherPtr = &x;

if(anotherPtr == ptr){

    cout<<"Same addresses are stored"<<endl;
}

else{
    cout<<"Different addresses are stored"<<endl;
}

int y = 6;

anotherPtr = &y;

if(*anotherPtr == *ptr){

    cout<<"Both addresses stored in pointer have same value"
<<endl;
}
else{

    cout<<"Both addresses stored in pointer have different values"

```

```
<<endl;
    }

    int *newPtr = new int;

    *newPtr = 1000;

    cout<<"Value pointed by newPtr: "<<*newPtr<<endl;
    cout<<"Address stored in newPtr: "<<newPtr<<endl;
    cout<<"Address of newPtr:"<< &newPtr<<endl;

    delete newPtr;

    newPtr = nullptr;
    cout<<"Value pointed by newPtr: "<<*newPtr<<endl;

    */

    return 0;
}
```