Q1. What will be the output of the given programs? In case of an error(s), circle that part in the code and mention the reason for that error(s) in one line [2 x 9 = 18 marks]

Assume code is written inside the main function for all 9 parts of this question.

1	`int my_var = 15;`
	`int my_var2 = 20;`
	
	`int* my_var_ptr = &my_var;`
	`int* my_ptr = my_var_ptr;`
	`my_var_ptr = my_var2;`
	
	`cout << *my_var_ptr;`
		ERROR: cannot save integer value in pointer

2	`double a = 12.4;`
	`double* ptr = &a;`
	`double b = 22.5;`
	`double* ptr2 = &b;`
	`*ptr = b;`
	
	`if (ptr == ptr2)`
	`  cout << "equal";`
	`else`
	`  cout << "not equal";`
		not equal

3	`bool x = 0;`
	`int y = 19;`
	`char z = 's';`
	
	`int* i = &y;`
	`char* c = &y;`
	`void* ptr = &z;`
	
	`cout << *ptr;`
		ERROR: cannot save integer address in character pointer

4	`bool x = 0;`
	`int y = 19;`
	`int* yy = &y;`
	`short* z = (short*)yy;`
	
	`cout << --(*z) << endl;`
	`cout << ++(*yy);`
		18
		19

5	`char my_array[] = "OOP is so easy!";`
	`char* my_ptr;`
	`my_ptr = my_array;`
	`my_ptr = my_ptr + 13;`
		yes

	cout << *my_ptr; cout << *(my_ptr - 3); my_ptr--; cout << *my_ptr;	

6	float data[] = { 10.2,20.0, 30.5, 40.5, 76.1};
	
	double * a = new double;
	*a = *(data + 2);
	a++;
	
	*a = (*a - *(a - 1));
	cout << *(data+3);
		40.5

7	char **s = new char*[2];
	for(int i=0; i<4; i++)
	*(s+i) = new char[2];
	
	((s+1)) = 65; //ASCII for 'A'
	*(*s+1) = 66;
	s[1][1] = 67;
	**s = 68;
	
	for(int j=0; j<2; j++)
	{
	for(int k=0; k<2; k++){
	cout<< s[j][k]<<" ";
	}
	cout<<endl;
	}
		D B
		A C

8	int var1 = 170;
	int *p= &var1;
	const int* ptr = p;
	
	cout<< *p <<" "<< *ptr <<endl;
	cout<< (*p)++ <<" "<< ++(*ptr);
		ERROR: ptr is a READ-ONLY pointer to constant

9	const int x = 11;
	const int *const ptr = &x;
	int y = 15;
	const int *p = &x;
	int * const ptr2 = &y;
	ptr2 = p;
	
	cout<<*p<<" "<<*ptr<<" "<<*ptr2;
		ERROR: cannot change value of ptr2, it is a constant pointer

Q2. Rewrite the given code using **dynamic 2D arrays**. Only use deference operator (*) to access or modify array elements. [5 marks]

```
int func(int arr[][3],int r, int c){

    int sum=0;
    for(int i=0; i<r; i++)
    {
        for(int j=0; j<c; j++){

            if(i == j)
              sum += arr[i][j];
        }
    }
    return sum;
}

int main()
{
    int row=3, col=3;
    int a[][3] =
{1,2,3,4,5,6,7,8,9};
    cout<<func(a, row, col)<<endl;

    return 0;
}
```

```
int func (int **arr, int r, int
c){

int sum=0;
    for(int i=0; i<r; i++)
    {
     for(int j=0; j<c; j++){

       if(i == j)
         sum += *((*arr + i) +j);

       }
    }
    return sum;
}

int main()
{
    int row=3, col=3;
    int **a=new int*[row];

    for(int i=0;i<row; i++){
        *(a+i)=new int[col];
    }

    int count=1;
    for(int i=0; i<row; i++)
    {
     for(int j=0; j<col; j++){

       *((*arr + i) +j)=count++;

       }
    }

 cout<<func(a, row, col)<<endl;

    return 0;
}
```

1 mark for all lines in red
0.5 if partially correct

What would be the output produced by executing the following C++ codes? Identify errors, if any (either write output or error, both will not be accepted). All the code snippet contains #include<iostream> and using namespace std;

a. [3 Mark]

```
int main( )
{
   void* vp;
   char ch = 'g'; const char *cp = "goofy";
   int j = 20;
   vp = &ch;
   cout << *(char*)vp;
   vp = &j;
   cout << *(int*)vp;
   vp =(void*) cp;
   cout << (char*)vp + 3 << endl;
   return 0;
}
```

Output/Errors:

g20fy

b. [8 Mark]

```
const char* c[] = { "Oopsmid-1", "MID", "OOP", "Exam" };
char const ** cp[] = { c + 3, c + 2, c + 1, c };
char const *** cpp = cp;

int main()
{
   cout << **(cpp + 1) << endl;
   cout << *(*(*(cpp + 2) + 2) + 3) << endl;
   cout << *((*cpp) - 2) << endl;
   cout << *(*(cpp + 3) + 0) + 3 << endl;
   return 0;
}
```

Output/Errors:

OOP
m
MID
smid-1