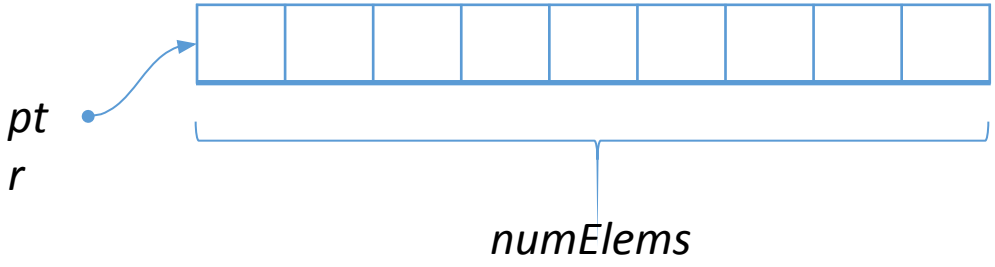
Lecture-28

Class – conversion constructor and *explicit* keyword

class - arr

```
4   class arr{
5       int *ptr;
6       int numElems;
7   public:
8       arr();        // default constructor
9       arr(int);     // single parameter constructor
10      ~arr();       // default destructor
11
12      friend void printArr(const arr&);
13  };
```



class - arr

```
arr::arr(){
    ptr = nullptr;
    numElems = 0;
}

arr::arr(int n):numElems(n){
    ptr = new int[numElems]{0};
}

arr::~arr(){
    delete ptr;
}

void printArr(const arr& a){
    for (int i = 0; i < a.numElems; i++){
        cout << a.ptr[i] << " ";
    }
    cout << endl;
}
```

Default constructor that initial the pointer to null pointer and number of elements to zero

Single parameter constructor that sets number of elements to n (passed as parameter) and ptr to an array of n elements each initialized to zero.

Default destructor that deletes the pointer.

Friend function that prints the array.

class - arr

```
35  int main(){
36
37      arr a(10);
38      printArr(a);
39      printArr(5);
40
41  }
```

Declare an array of 10 elements.

Print the array ‘a’.

Print the integer 5??

```
C:\Users\cveer\Documents\explicit.exe
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0

Process returned 0 (0x0)   execution time : 0.313 s
Press any key to continue.
```

In the function call *printArr*(5), the compiler is using the single parameter constructor as a conversion constructor to convert integer 5 to an ***arr*** object of 5 elements. This conversion is implicit and may be unintended.

class - arr

```
4   class arr{
5       int *ptr;
6       int numElems;
7   public:
8       arr();          // default constructor
9       explicit arr(int);    // single parameter constructor
10      ~arr();         // default destructor
11
12      friend void printArr(const arr&);
13  };
```

To avoid any implicit conversion, we make single parameter constructor *explicit*.

class - arr

```
35  int main(){
36
37      arr a(10);
38      printArr(a);
39      printArr(5);
40
41  }
42
43
```

Now the compiler is not able to locate a suitable function to call and gives an error message.

gs & others

Code::Blocks | Search results | Cccc | Build log | **Build messages** | CppCheck/Vera++ | CppCheck/Ve

File	Line	Message
		=== Build file: "no target" in "no project" (compiler: unknown) ===
C:\Users\cveer...		In function 'int main()':
C:\Users\cveer...	39	error: invalid initialization of reference of type 'const arr&' from expression of type 'int'
C:\Users\cveer...	28	note: in passing argument 1 of 'void printArr(const arr&)'
		=== Build failed: 1 error(s), 0 warning(s) (0 minute(s), 2 second(s)) ===