











Array Access

- X=ar[2];
- ar[3]=2.7;
- What is the differenc ebetween ar[i]++, ar[i++], ar[++i];

















Pointers				
 identifier of an array is equivalent to the address of its first element 				
 int numbers [20]; int * p; 				
p = numbers // Valid numbers = p // Invalid				
 p and numbers are equivalent and they have the same properties 				
 Only difference is that we could assign another value to the pointer p whereas numbers will always point to the first of the 20 integer numbers of type int 				

Pointer Arithmetic

- int *x, *y
- int z;
- Can do
 - − z=x-y;
 - x=NULL;
 - if(c==NULL)
 - Also, what is void * ?













Multidimensional arrays					
 #include <stdio.h></stdio.h> 					
int main() {		36			
 float *pf; 		0.4000	0.5000	0.6000	
• float m[][3]={ {0.	1, 0.2, 0.3},	0.6000	0.5000	0.4000	
• {0.4, 0.5, 0.6},					
• {0.7, 0.8, 0.9}};					
 printf("%d \n",sizeof(m)); 					
 pf=m[1]; 					
 printf("%f %f %f \ 	 printf("%f %f %f \n",*pf, *(pf+1), *(pf+2)); 				
 printf("%f %f %f \ 	n",*pf, *(pf++), *(p	of++));			
• }					







strcpy

```
void strcpy(char *s, char *t) {
  while( (*s = *t) != `\0' ) {
      s++;
      t++;
      }
  }
}
```



EX

Char (*p2)[100]; char name[100]; char *p1; p1=name; p2=name; // What is the difference? Consider p1+1 and p2+1



Command-Line Arguments



Pointers to Functions

- Although functions are not variables, it is possible to assign a pointer to a function.
- That pointer could be manipulated, assigned, placed on arrays, or passed/returned to/by functions.







