Summary and plan

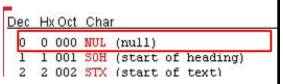
- **(Primitive) Types and sizes**
  - **Types:** **char, short, int, long, unsigned short, unsigned int, float, double …..**
  - **Constant values (literals)**
    - **char**
    - **int**
    - **float**
- Array and "**strings**"

today

- Expressions
  - Basic operators
  - Type promotion and conversion
  - Other operators
  - Precedence of operators

100

Strings ⇔ Character Arrays !



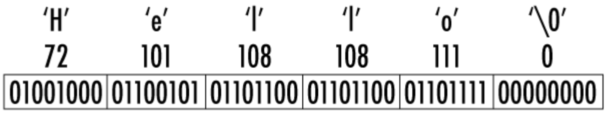
```
Dec  Hx Oct  Char
0    0 000 NUL (null)
1    1 001 SOH (start of heading)
2    2 002 STX (start of text)
```

- There is no separate "string" type in C
- Strings are just arrays of char that end with `'\0'`
  - `char s[]= "Hello";`

⇩

`'H'`	`'e'`	`'l'`	`'l'`	`'o'`	`'\0'`

`'\0'` added for you



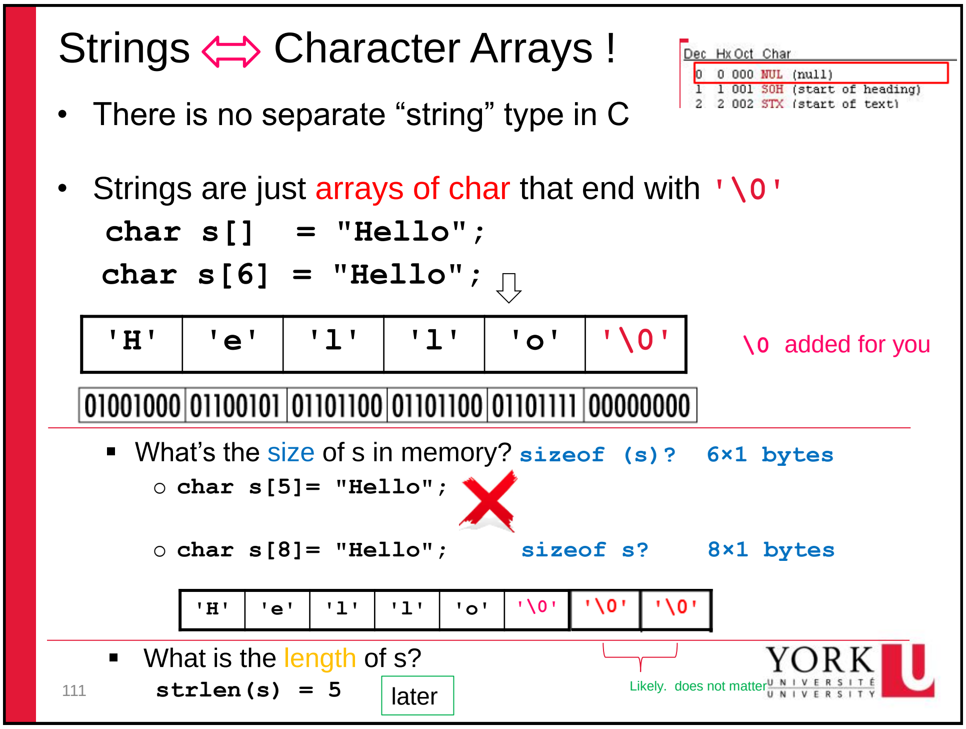
'H'	'e'	'l'	'l'	'o'	'\0'
72	101	108	108	111	0
01001000	01100101	01101100	01101100	01101111	00000000

is equivalent to

```
char s[]= {'H', 'e', 'l', 'l', 'o', '\0'}
```

110

no \0?



Strings ⟺ Character Arrays !

Dec	Hx	Oct	Char	
0	0	000	NUL	(null)
1	1	001	SOH	(start of heading)
2	2	002	STX	(start of text)

- There is no separate "string" type in C
- Strings are just arrays of char that end with `'\0'`

```
char s[]  = "Hello";
char s[6] = "Hello";
```

'H'	'e'	'l'	'l'	'o'	'\0'
01001000	01100101	01101100	01101100	01101111	00000000

\0 added for you

- What's the size of s in memory? `sizeof (s)?` 6×1 bytes
  - ○ `char s[5]= "Hello";` ✗
  - ○ `char s[8]= "Hello";` `sizeof s?` 8×1 bytes

'H'	'e'	'l'	'l'	'o'	'\0'	'\0'	'\0'

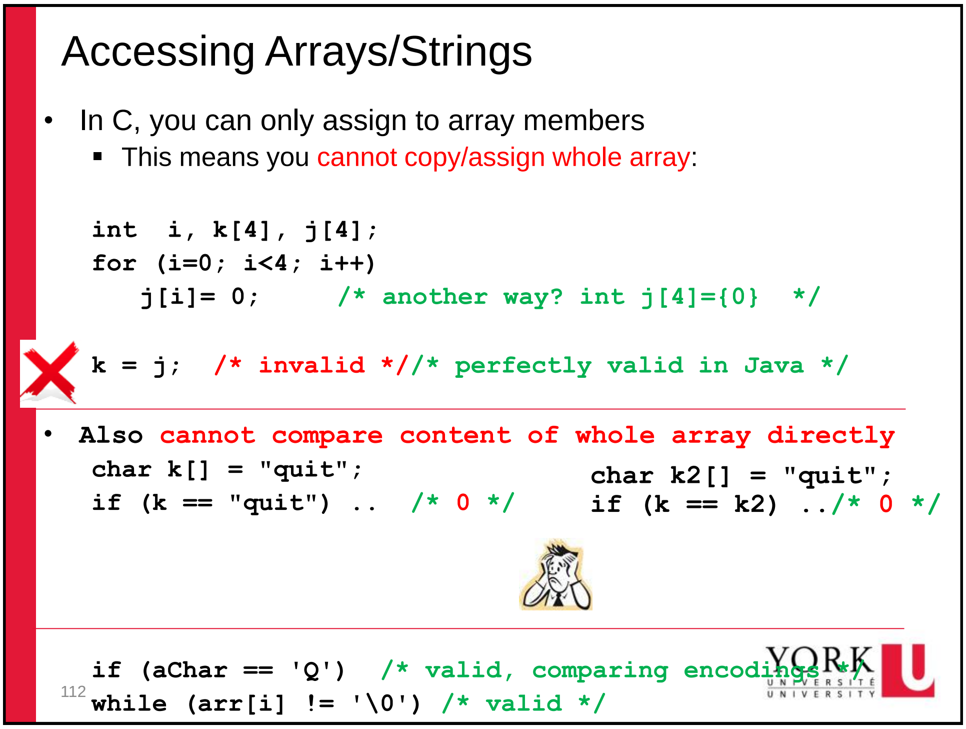
Likely. does not matter

- What is the length of s?
  `strlen(s) = 5`

later

YORK UNIVERSITÉ UNIVERSITY

111



Accessing Arrays/Strings

- In C, you can only assign to array members
  - This means you cannot copy/assign whole array:

```
int  i, k[4], j[4];
for (i=0; i<4; i++)
   j[i]= 0;     /* another way? int j[4]={0}  */

k = j;  /* invalid *//* perfectly valid in Java */
```

- Also cannot compare content of whole array directly

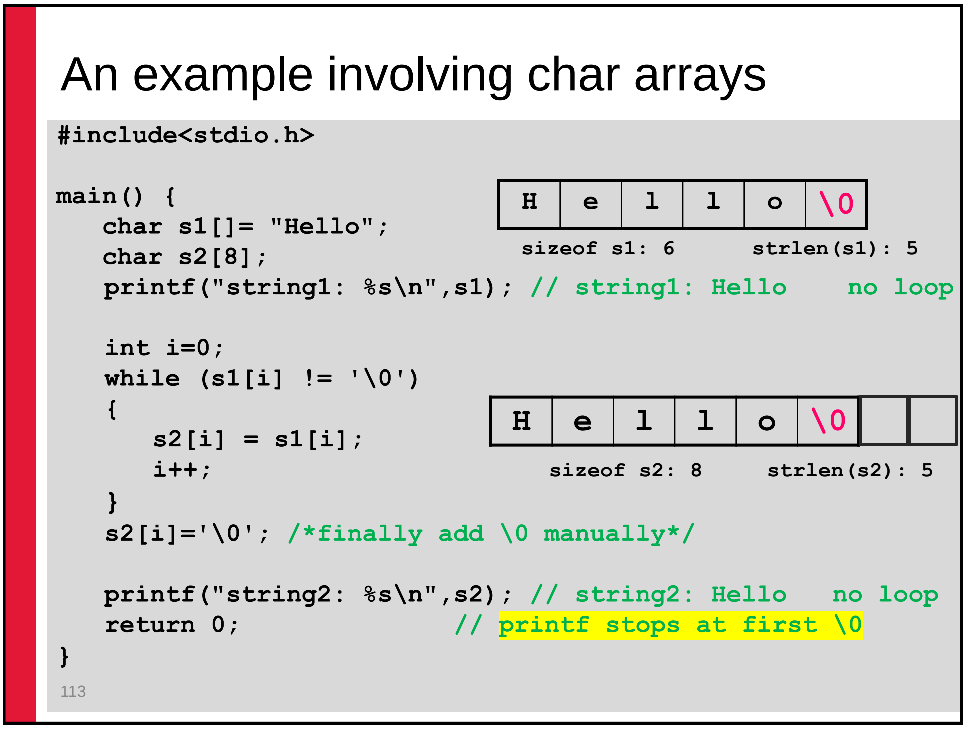
```
char k[] = "quit";
if (k == "quit") ..  /* 0 */
```

```
char k2[] = "quit";
if (k == k2) ../* 0 */
```

```
if (aChar == 'Q')  /* valid, comparing encodings */
while (arr[i] != '\0') /* valid */
```

YORK UNIVERSITÉ UNIVERSITY

112



An example involving char arrays

```
#include<stdio.h>

main() {
    char s1[]= "Hello";
    char s2[8];
    printf("string1: %s\n",s1); // string1: Hello     no loop

    int i=0;
    while (s1[i] != '\0')
    {
        s2[i] = s1[i];
        i++;
    }
    s2[i]='\0'; /*finally add \0 manually*/

    printf("string2: %s\n",s2); // string2: Hello   no loop
    return 0;                // printf stops at first \0
}
```

H	e	l	l	o	\0

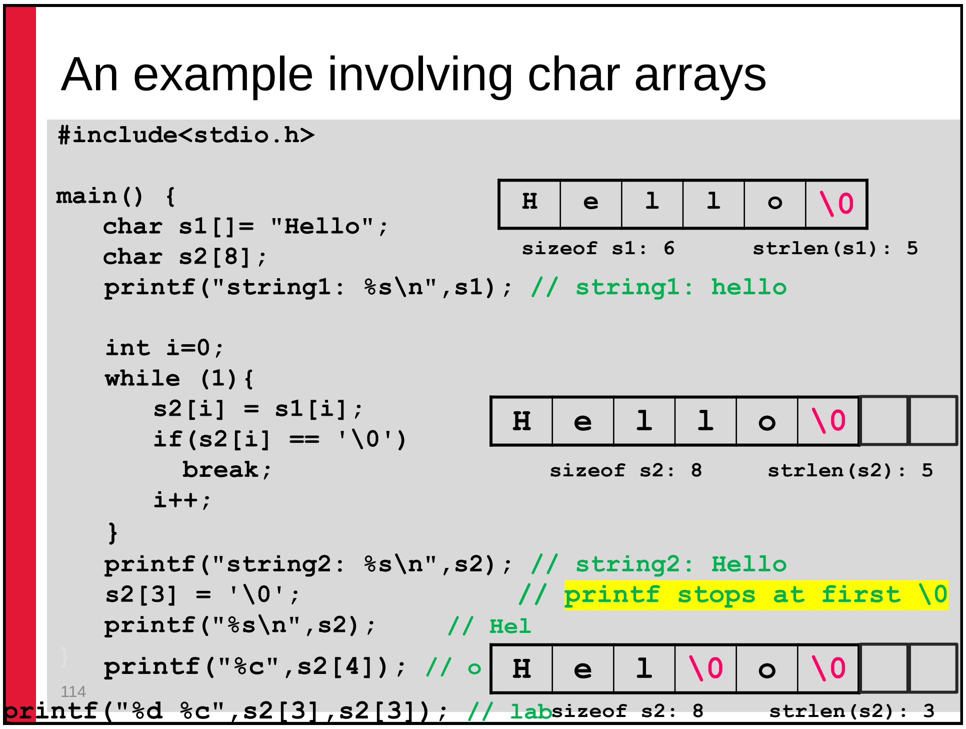
sizeof s1: 6 strlen(s1): 5

H	e	l	l	o	\0		

sizeof s2: 8 strlen(s2): 5

113

113



An example involving char arrays

```
#include<stdio.h>

main() {
    char s1[]= "Hello";
    char s2[8];
    printf("string1: %s\n",s1); // string1: hello

    int i=0;
    while (1){
        s2[i] = s1[i];
        if(s2[i] == '\0')
          break;
        i++;
    }
    printf("string2: %s\n",s2); // string2: Hello
    s2[3] = '\0';              // printf stops at first \0
    printf("%s\n",s2);      // Hel
}
    printf("%c",s2[4]); // o
printf("%d %c",s2[3],s2[3]); // lab
```

H	e	l	l	o	\0

sizeof s1: 6 strlen(s1): 5

H	e	l	l	o	\0		

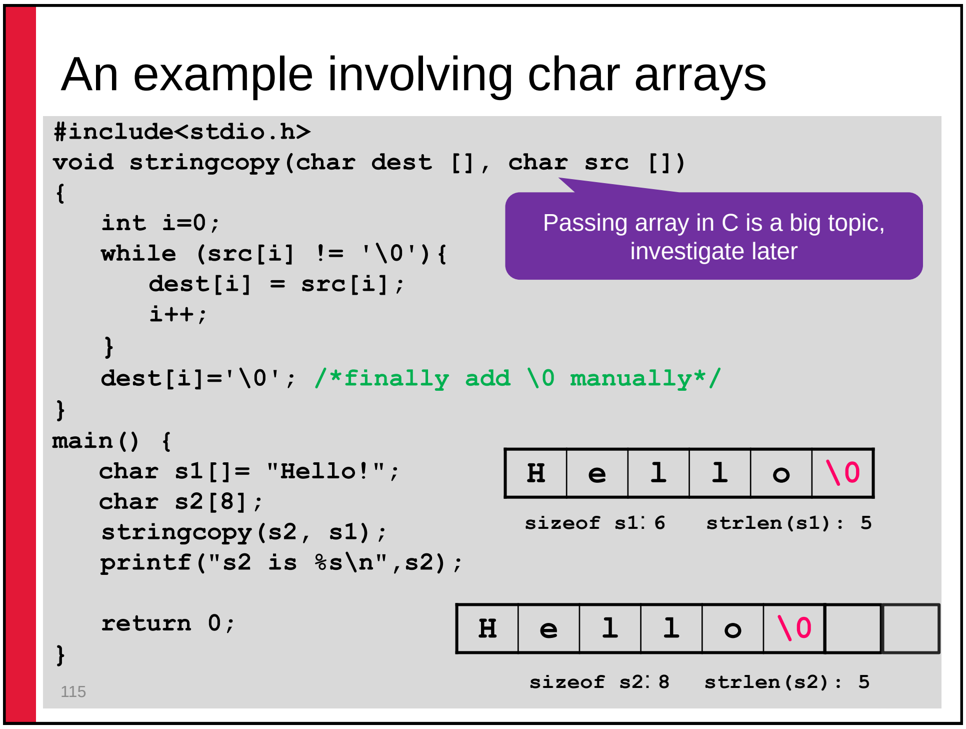
sizeof s2: 8 strlen(s2): 5

H	e	l	\0	o	\0		

sizeof s2: 8 strlen(s2): 3

114

114



An example involving char arrays

```
#include<stdio.h>
void stringcopy(char dest [], char src [])
{
   int i=0;
   while (src[i] != '\0'){
      dest[i] = src[i];
      i++;
   }
   dest[i]='\0'; /*finally add \0 manually*/
}
main() {
   char s1[]= "Hello!";
   char s2[8];
   stringcopy(s2, s1);
   printf("s2 is %s\n",s2);

   return 0;
}
```

Passing array in C is a big topic, investigate later

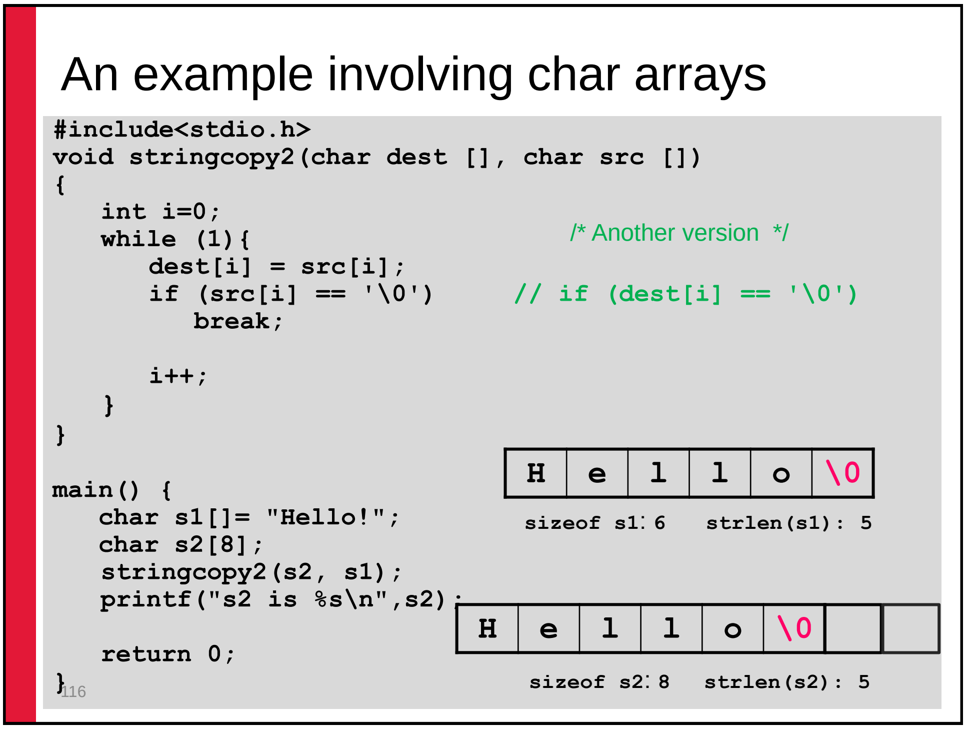
H	e	l	l	o	\0

sizeof s1: 6   strlen(s1): 5

H	e	l	l	o	\0		

sizeof s2: 8   strlen(s2): 5

115



An example involving char arrays

```
#include<stdio.h>
void stringcopy2(char dest [], char src [])
{
   int i=0;
   while (1){                      /* Another version  */
      dest[i] = src[i];
      if (src[i] == '\0')       // if (dest[i] == '\0')
         break;

      i++;
   }
}

main() {
   char s1[]= "Hello!";
   char s2[8];
   stringcopy2(s2, s1);
   printf("s2 is %s\n",s2);

   return 0;
}
```

H	e	l	l	o	\0

sizeof s1: 6   strlen(s1): 5

H	e	l	l	o	\0		

sizeof s2: 8   strlen(s2): 5

116