

**Higher School of Applied Sciences of Tlemcen**  
**1-year computer science course**



# **Pointers**

## **Part 1**

**Presented by:**  
**Dr. Imane NEDJAR**

# **Part 1**

## **Overview of Pointers**

---

## Definition

A pointer is a special type of variable that stores the memory address of another variable

Variable  
Int

Variable  
Float

Variable  
char

Arrays

```
int x=19
```

The address of X is **0x7ffcfec54b1**

Memory address

0x7ffcfec54b0

0x7ffcfec54b1

0x7ffcfec54b2

0x7ffcfec54b3

0x7ffcfec54b4

0x7ffcfec54b5

0x7ffcfec54b6

0x7ffcfec54b7

0x7ffcfec54b8

0x7ffcfec54b9

0x7ffcfec54c0

19

X

```
int x=19
```

L'adresse de X est **0x7ffcfec54b1**

```
int *p
```

```
P=&x
```

```
*p=19
```

Memory address

0x7ffcfec54b0

0x7ffcfec54b1

0x7ffcfec54b2

0x7ffcfec54b3

0x7ffcfec54b4

0x7ffcfec54b5

0x7ffcfec54b6

0x7ffcfec54b7

0x7ffcfec54b8

0x7ffcfec54b9

0x7ffcfec54c0

19

x

0x7ffcfec54b1

p

# Pointers

## Pointers and Integers

```
int x=19
```

```
int y=20
```

```
int *p
```

```
p=&x
```

```
*p=19
```

```
int *q
```

```
q=&y
```

```
*q=20
```

Memory address

0x7ffcfe54b0	20	y
0x7ffcfe54b1	19	x
0x7ffcfe54b2		
0x7ffcfe54b3		
0x7ffcfe54b4		
0x7ffcfe54b5		
0x7ffcfe54b6	0x7ffcfe54b1	p
0x7ffcfe54b7		
0x7ffcfe54b8		
0x7ffcfe54b9		
0x7ffcfe54c0	0x7ffcfe54b0	q

**int x=19**

**int y=20**

**int \*p**

**p=&x**

**\*p=19**

**int z**

**int \*q**

**q=&y**

**\*q=20**

**int \*t**

**t=&z**

**\*t=\*p+\*q=39**

Memory address

0x7ffcfe54b0

20

y

0x7ffcfe54b1

19

x

0x7ffcfe54b2

0x7ffcfe54b3

39

z

0x7ffcfe54b4

0x7ffcfe54b5

0x7ffcfe54b6

0x7ffcfe54b1

p

0x7ffcfe54b7

0x7ffcfe54b8

0x7ffcfe54b9

0x7ffcfe54b3

t

0x7ffcfe54c0

0x7ffcfe54b0

q

```
printf("%d",x)
```

-> 19

```
printf("%d",*p)
```

-> 19

```
printf("%p",p)
```

-> 0x7ffcfec54b1

Memory address

0x7ffcfec54b0	20	y
0x7ffcfec54b1	19	x
0x7ffcfec54b2		
0x7ffcfec54b3		z
0x7ffcfec54b4		
0x7ffcfec54b5		
0x7ffcfec54b6	0x7ffcfec54b1	p
0x7ffcfec54b7		
0x7ffcfec54b8		
0x7ffcfec54b9	0x7ffcfec54b3	t
0x7ffcfec54c0	0x7ffcfec54b0	q



**float x=19.5**

L'adresse de X est **0x7ffcfec54b1**

Memory address

0x7ffcfec54b0

0x7ffcfec54b1

0x7ffcfec54b2

0x7ffcfec54b3

0x7ffcfec54b4

0x7ffcfec54b5

0x7ffcfec54b6

0x7ffcfec54b7

0x7ffcfec54b8

0x7ffcfec54b9

0x7ffcfec54c0

**19.5**

**X**

**float x=19.5**

L'adresse de X est **0x7ffcfec54b1**

**float \*p**

**P=&x**

**\*p=19.5**

Memory address

0x7ffcfec54b0

0x7ffcfec54b1

0x7ffcfec54b2

0x7ffcfec54b3

0x7ffcfec54b4

0x7ffcfec54b5

0x7ffcfec54b6

0x7ffcfec54b7

0x7ffcfec54b8

0x7ffcfec54b9

0x7ffcfec54c0

**19.5**

**x**

**0x7ffcfec54b1**

**p**

**float x=19.5**

**float y=20.2**

**float \*p      float \*q**

**p=&x      q=&y**

**\*p=19.5      \*q=20.2**

Memory address

0x7ffcfe54b0	20.2	y
0x7ffcfe54b1	19.5	x
0x7ffcfe54b2		
0x7ffcfe54b3		
0x7ffcfe54b4		
0x7ffcfe54b5		
0x7ffcfe54b6	0x7ffcfe54b1	p
0x7ffcfe54b7		
0x7ffcfe54b8		
0x7ffcfe54b9		
0x7ffcfe54c0	0x7ffcfe54b0	q

**float x=19.5**

**float y=20.2    float z**

**float \*p    float \*q    float \*t**

**p=&x    q=&y    t=&z**

**\*p=19.5    \*q=20.2**

**\*t=\*p+\*q=39.7**

Memory address

0x7ffcfc54b0	20.2	y
0x7ffcfc54b1	19.5	x
0x7ffcfc54b2		
0x7ffcfc54b3	39.7	z
0x7ffcfc54b4		
0x7ffcfc54b5		
0x7ffcfc54b6	0x7ffcfc54b1	p
0x7ffcfc54b7		
0x7ffcfc54b8		
0x7ffcfc54b9	0x7ffcfc54b3	t
0x7ffcfc54c0	0x7ffcfc54b0	q

```
printf("%f",x)
```

-> 19.5

```
printf("%f",*p)
```

-> 19.5

```
printf("%p",p)
```

-> 0x7ffcfec54b1

Memory address

0x7ffcfec54b0	20.2	y
0x7ffcfec54b1	19.5	x
0x7ffcfec54b2		
0x7ffcfec54b3		z
0x7ffcfec54b4		
0x7ffcfec54b5		
0x7ffcfec54b6	0x7ffcfec54b1	p
0x7ffcfec54b7		
0x7ffcfec54b8		
0x7ffcfec54b9	0x7ffcfec54b3	t
0x7ffcfec54c0	0x7ffcfec54b0	q

# Pointers

## Pointers and Characters

**char x='A'**

L'adresse de X est **0x7ffcfec54b1**

Memory address

0x7ffcfec54b0

0x7ffcfec54b1

0x7ffcfec54b2

0x7ffcfec54b3

0x7ffcfec54b4

0x7ffcfec54b5

0x7ffcfec54b6

0x7ffcfec54b7

0x7ffcfec54b8

0x7ffcfec54b9

0x7ffcfec54c0

'A'

X

# Pointers

## Pointers and Characters

**char x='A'**

L'adresse de X est **0x7ffcfec54b1**

**char \*p**

**P=&x**

**\*p='A'**

Memory address

0x7ffcfec54b0

0x7ffcfec54b1

0x7ffcfec54b2

0x7ffcfec54b3

0x7ffcfec54b4

0x7ffcfec54b5

0x7ffcfec54b6

0x7ffcfec54b7

0x7ffcfec54b8

0x7ffcfec54b9

0x7ffcfec54c0

'A'

0x7ffcfec54b1

x

p

# Pointers

## Pointers and Characters

**char x='A'**

**char y='B'**

**char \*p      char \*q**

**p=&x      q=&y**

**\*p='A'      \*q='B'**

Memory address

0x7ffcfc54b0	'B'	y
0x7ffcfc54b1	'A'	x
0x7ffcfc54b2		
0x7ffcfc54b3		
0x7ffcfc54b4		
0x7ffcfc54b5		
0x7ffcfc54b6	0x7ffcfc54b1	p
0x7ffcfc54b7		
0x7ffcfc54b8		
0x7ffcfc54b9		
0x7ffcfc54c0	0x7ffcfc54b0	q



# Pointers

## Pointers and Characters

```
printf("%c",x)
```

-> 'A'

```
printf("%c",*p)
```

-> 'A'

```
printf("%p",p)
```

-> 0x7ffcfec54b1

Memory address

0x7ffcfec54b0	'B'	y
0x7ffcfec54b1	'A'	x
0x7ffcfec54b2		
0x7ffcfec54b3		
0x7ffcfec54b4		
0x7ffcfec54b5		
0x7ffcfec54b6	0x7ffcfec54b1	p
0x7ffcfec54b7		
0x7ffcfec54b8		
0x7ffcfec54b9		
0x7ffcfec54c0	0x7ffcfec54b0	q

# Pointers

## Pointers and Characters

### Note:

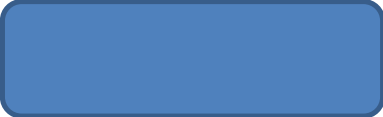
Any modification to **x** will also modify **\*P**,  
and vice versa

# Pointers

## Pointers and Characters

### Exercice 1

```
int myAge = 43;
```

```
 = &myAge;
```

### Exercise 2

**What does the program display?**

```
int myAge = 43;
```

```
int* ptr = &myAge;
```

```
printf("%d\n", myAge);
```

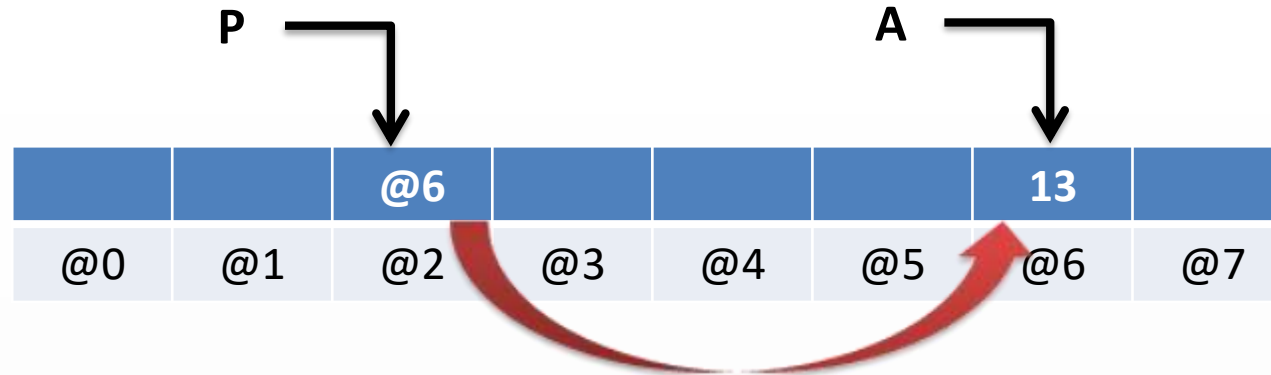
```
printf("%p\n", &myAge);
```

```
printf("%p\n", ptr);
```

# Pointers

## Pointers and Characters

### Exercise 3



Variable's name	Address	Value
A	<u>@6</u>	<u>13</u>
P	@2	@6
*P	<u>@6</u>	<u>13</u>

# Pointers

## Pointers and Characters

### Exercise 4

```
#include<stdio.h>
int i,j;
int *P1,*P2;
int main(){
i=2; j =5;
P1=&i; P2=&j;
(*P1)++;
*P2=*P1;
printf(" i=%d",i);
printf("j=%d",j);
printf(" *P1=%d",*P1);
printf(" *P2=%d",*P2);
return 0;}
```

