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

Pointers

Part 2

Presented by:
Dr. Imane NEDJAR

Part 2

Pointers and Arrays

Pointers and Arrays

An array's name is a constant pointer to its first element

- int tab[5];
- int *p;

p=tab -> p=&tab[0]

Pointers and Arrays

p=&tab[0]

p+1=&tab[1]

*(p+1)=tab[1]

- p+n=&tab[n]

Increment and Decrement

- p++=&tab[i+1]
- P--=&tab[i-1]

Pointers and Arrays

P ⇔(p+0) ⇔ tab[0];

*(P+1) ⇔tab [1];

*(P+2) ⇔tab [2];

……

*(P+i) ⇔tab [i];

……..

*(P+N-1) ⇔tab [N-1];

Note:

*P+1 ≠ *(P+1)

*P+1 : Add 1 to the value of the element pointed to by P

*(P+1): The value of the element pointed to the address following P, meaning P+1

Arrays of Integers

Reading an Array

```
for (i=0; i<N; i++)
        {scanf ("%d", &tab[i]);  }
```

```
for (p=tab; p<tab+N; p++)
        {scanf ("%d", p);}
```

Arrays of Integers

Displaying an Array

```
for (i=0; i<N; i++)
        {printf ("%d", tab[i]);  }
```

```
for (p=tab; p<tab+N; p++)
        {printf ("%d", *p);}
```

Arrays of Floats

Reading an Array

```
for (i=0; i<N; i++)
        {scanf ("%f", &tab[i]);  }
```

```
for (p=tab; p<tab+N; p++)
        {scanf ("%f", p);}
```

Arrays of Floats

Displaying an Array

```
for (i=0; i<N; i++)
        {printf ("%f", tab[i]);  }
```

```
for (p=tab; p<tab+N; p++)
        {printf ("%f", *p);}
```

Arrays of Characters

Reading an Array

```
for (i=0; i<N; i++)
        {scanf ("%c", &tab[i]);
         getchar();  }
```

```
for (p=tab; p<tab+N; p++)
        {scanf ("%c", p);
         getchar();  }
```

Arrays of Characters

Displaying an Array

```
for (i=0; i<N; i++)
        {printf ("%c", tab[i]);  }
```

```
for (p=tab; p<tab+N; p++)
        {printf ("%c", *p);}
```

Pointers and Arrays

Exercice

Provide the result of the following instruction

```
int T[10]={22,1,-3,5,0,3,11,8,4,20};
int *P;
P=T; printf(" %d",*P);  22
P=T; printf(" %d",*P+3); 25
P=T; printf(" %d",*(P+3)); 5
P=T; P++; printf(" %d",*P); 1
P=T; P++; printf(" %d",*(P+3)); 0
P=&T[0]; printf(" %d",*P); 22
P=&T[6]; printf(" %d",*P); 11
```