INFORMATICA I

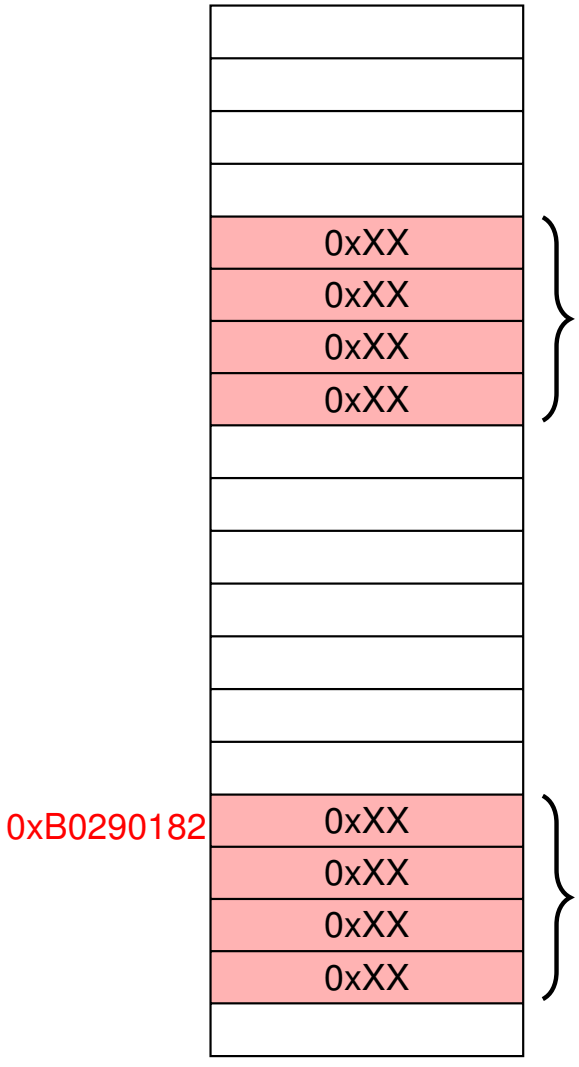
Incremento de punteros

Ing.Juan Carlos Cuttitta

Universidad Tecnológica Nacional
Facultad Regional Buenos Aires
Departamento de Ingeniería Electrónica

21 de mayo de 2019

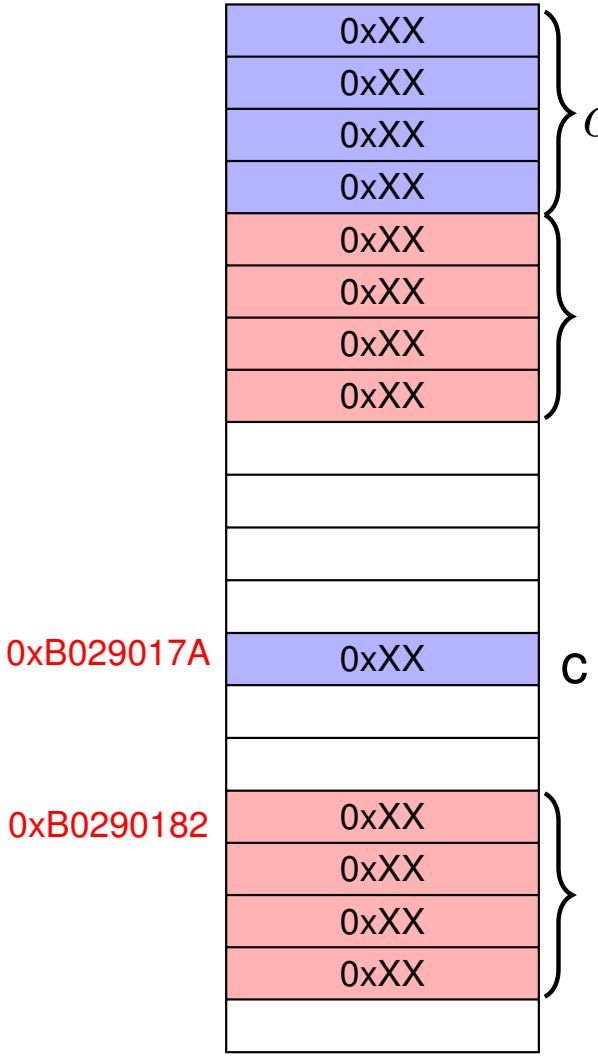
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3      int *intPtr, i;
4      char *charPtr, c;
5      i = 0x12345678;
6      c = 'N';
7      intPtr = &i;
8      charPtr = &c;
9      *charPtr= 'A';
10     *intPtr = 20;
11     charPtr++;
12     intPtr++;
13 }
```

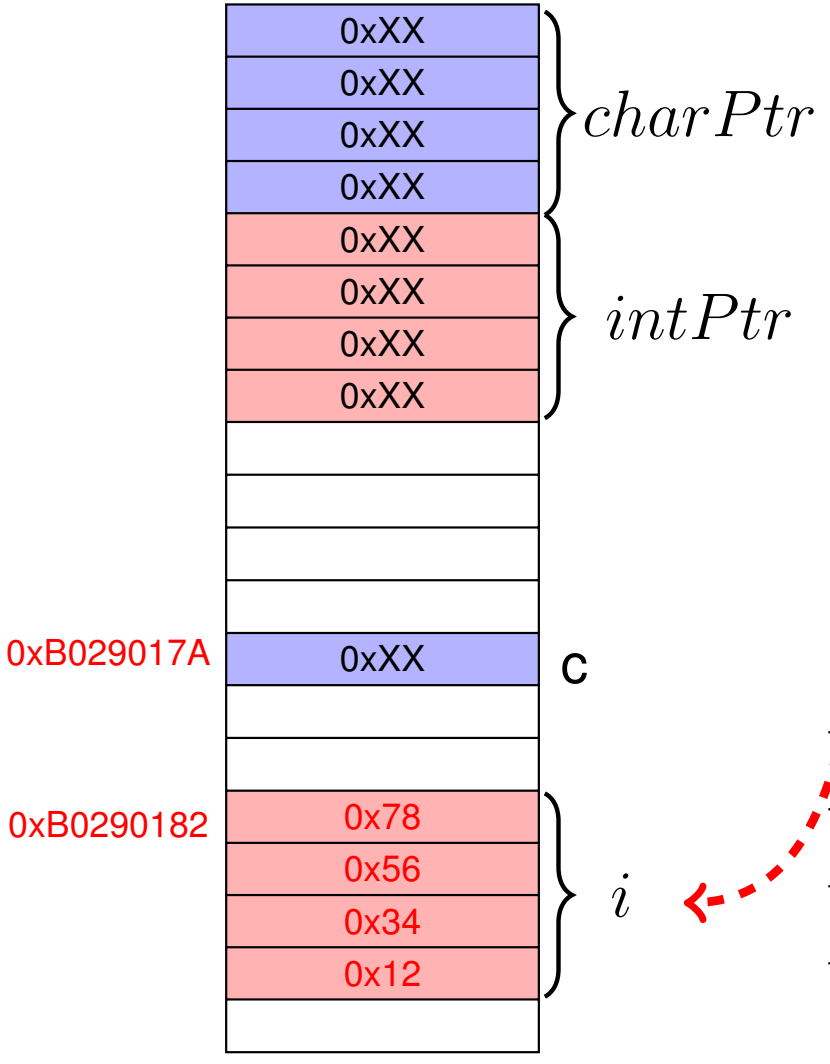
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3      int *intPtr, i;
4      char *charPtr, c;
5      i = 0x12345678;
6      c = 'N';
7      intPtr = &i;
8      charPtr = &c;
9      *charPtr= 'A';
10     *intPtr = 20;
11     charPtr++;
12     intPtr++;
13 }
```

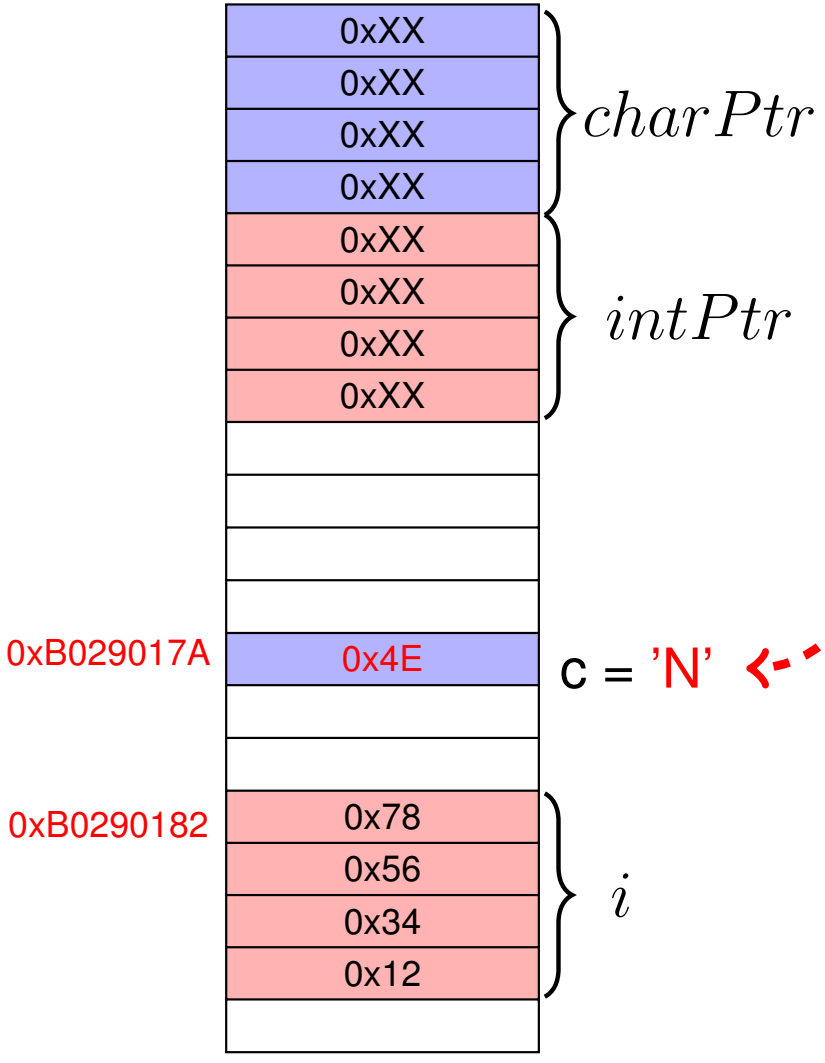
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr  = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13 }
```

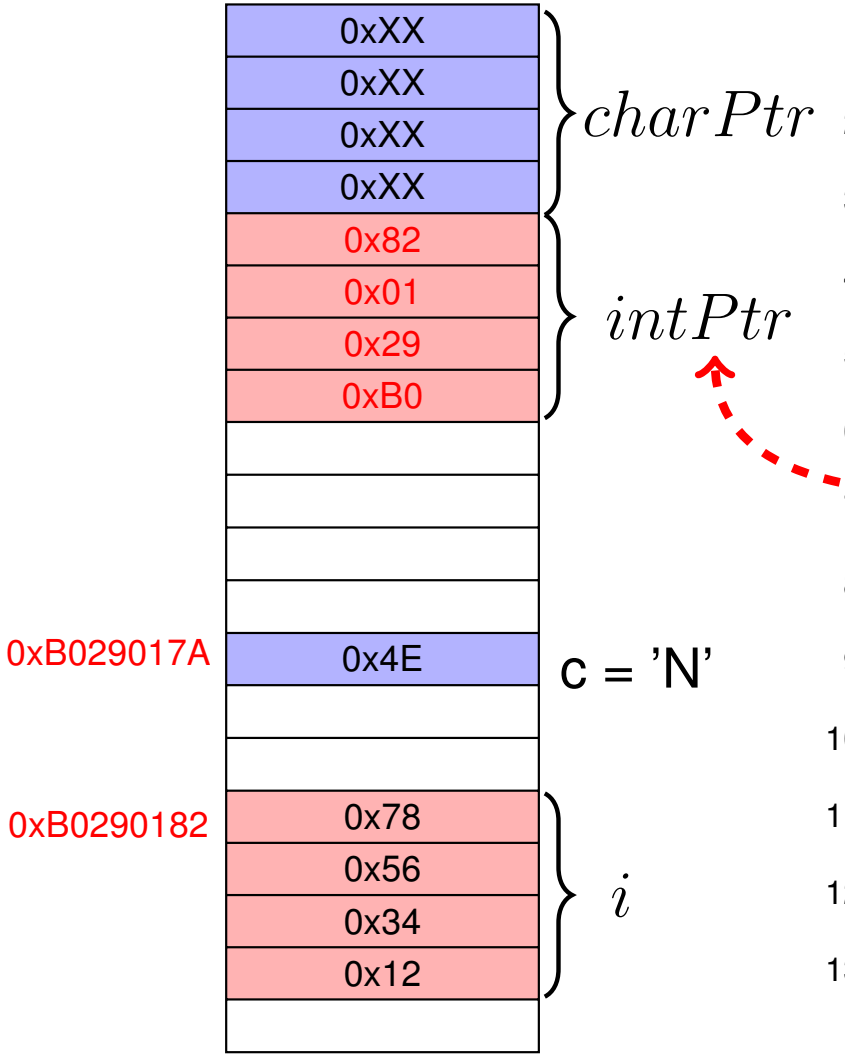
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr  = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13 }
```

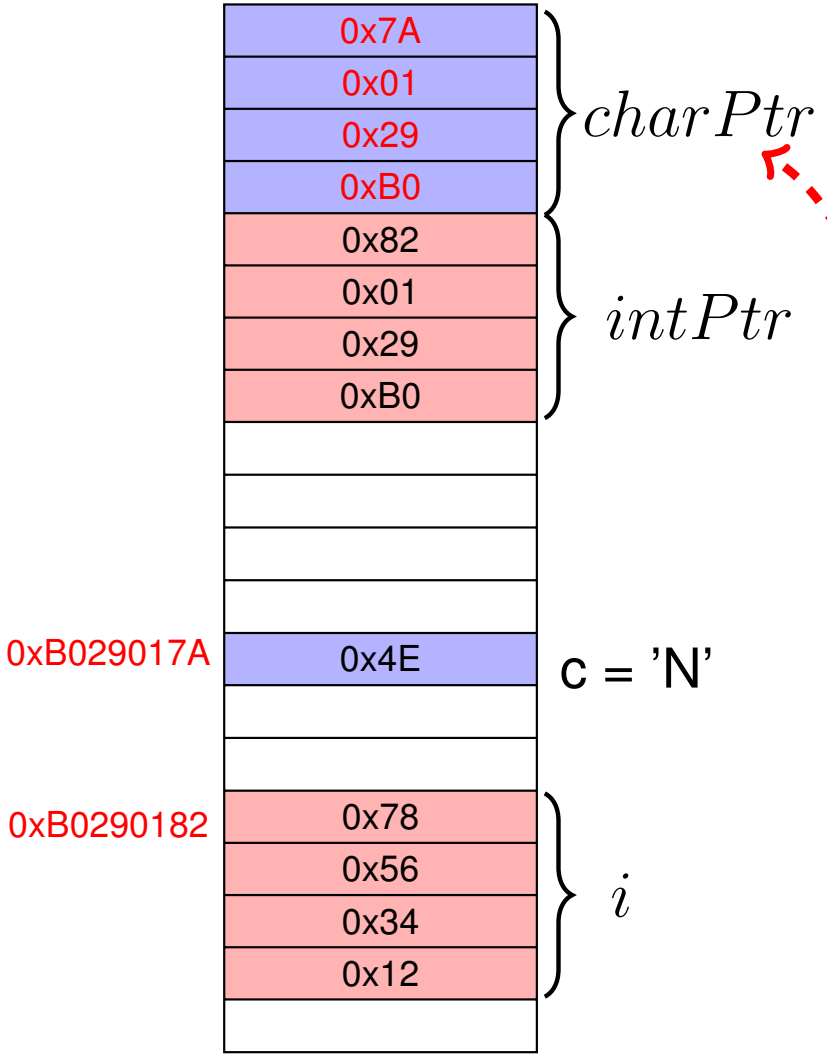
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13 }
```

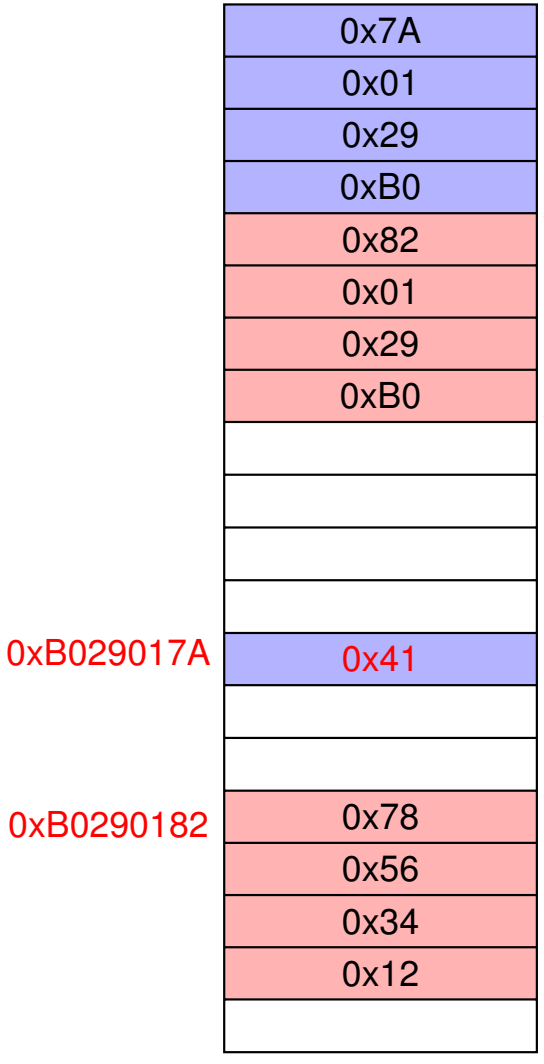
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr  = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13 }
```

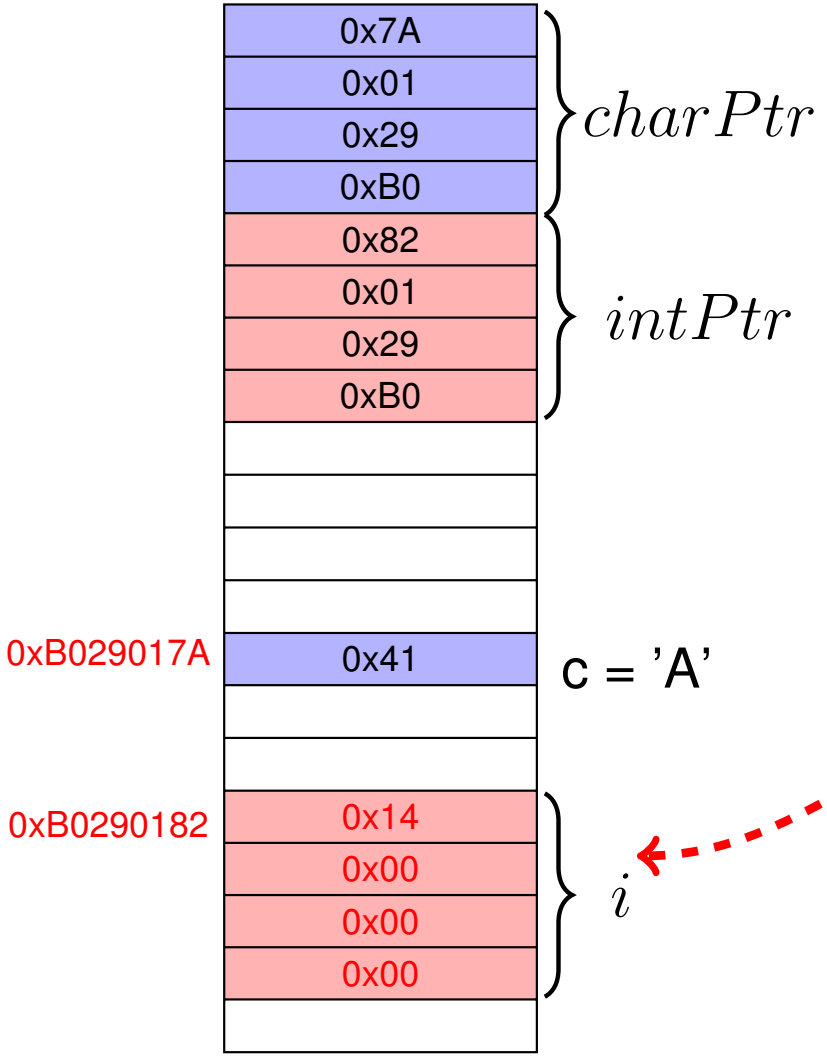
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr  = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13 }
```

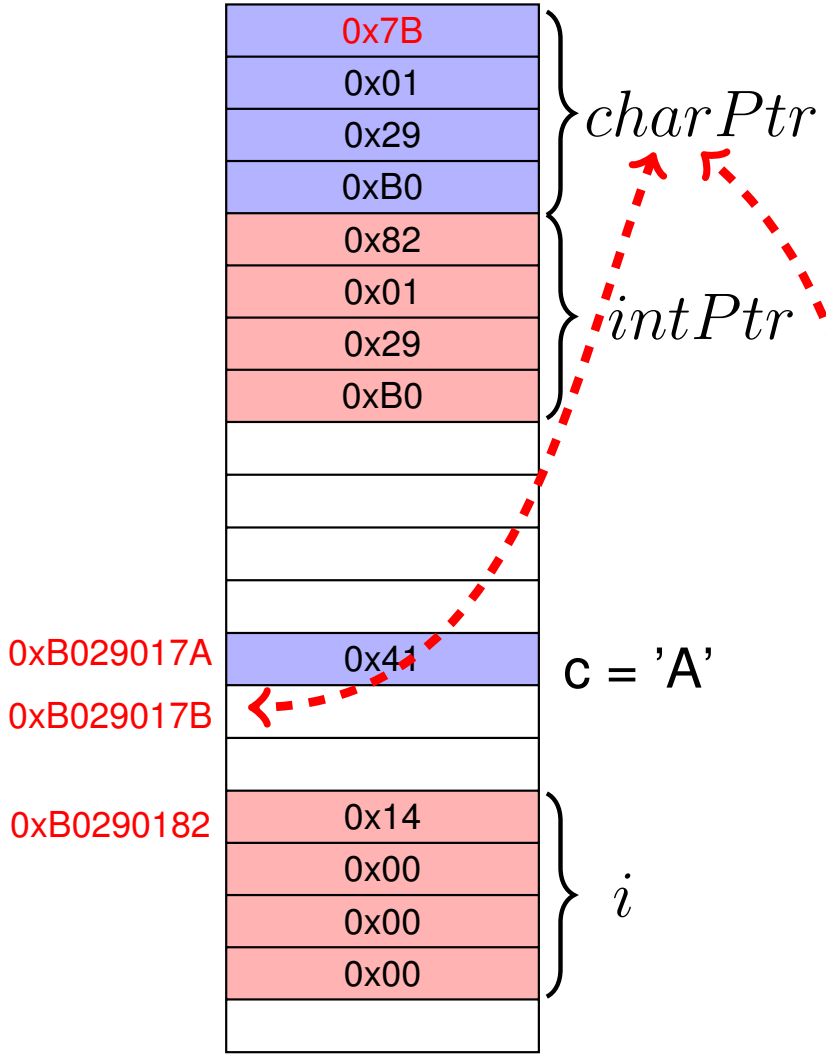
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr  = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13 }
```

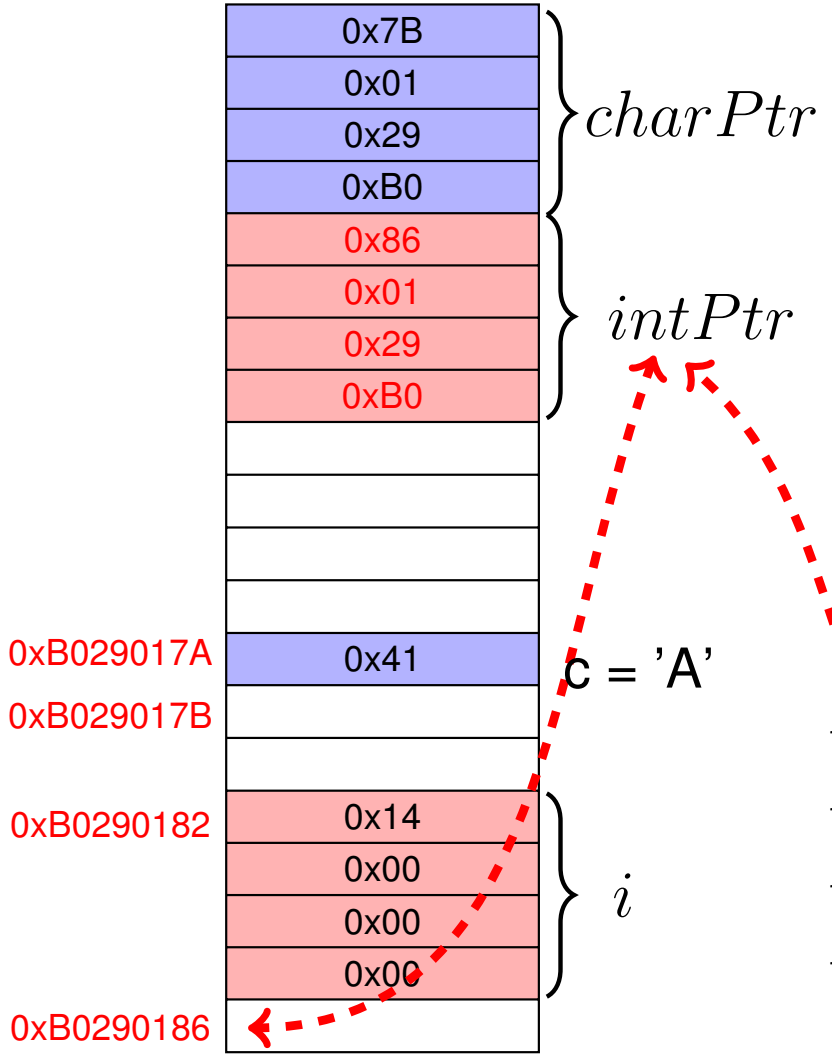
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr  = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13 }
```

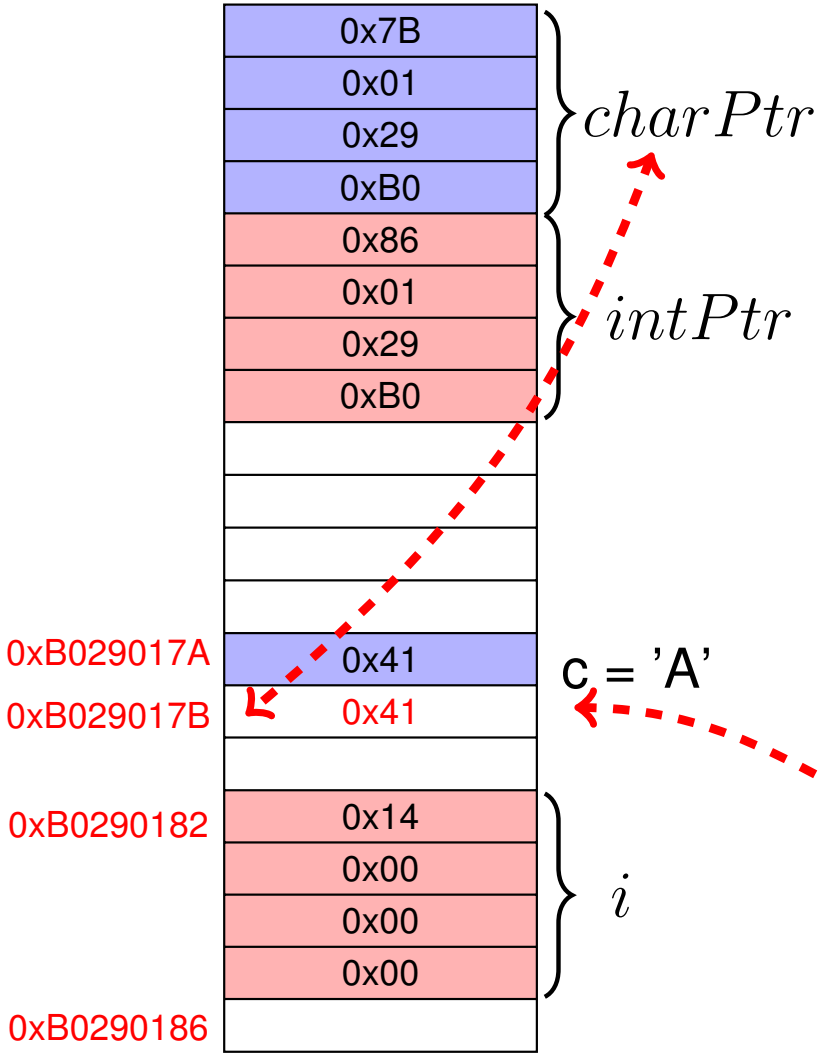
MEMORIA en ARQUITECTURA x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr  = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13 }
```

MEMORIA en *ARQUITECTURA* x86 32-bits



PROGRAMA FUENTE

```
1  void main ()
2  {
3     int *intPtr,i;
4     char *charPtr,c;
5     i = 0x12345678;
6     c = 'N';
7     intPtr  = &i;
8     charPtr = &c;
9     *charPtr= 'A';
10    *intPtr = 20;
11    charPtr++;
12    intPtr++;
13    *charPtr= 'A';
14 }
```

Algunas reglas para recordar !!!

int A;
int $*p$;

$$\underbrace{p}_{int*} = \underbrace{\&A}_{int*}; \quad \overbrace{int}$$

$$\underbrace{A}_{int} = \underbrace{7}_{int};$$

$$\underbrace{*p}_{int} = \underbrace{7}_{int}; \quad \overbrace{int*}$$

int $V[\,] = \{10, 20\}$;
int $*p$;

$$\underbrace{p}_{int*} = \underbrace{\&V[0]}_{int*}; \quad \overbrace{int}$$

$$\underbrace{V[1]}_{int} = \underbrace{7}_{int};$$

$$\underbrace{*(p+1)}_{int} = \underbrace{7}_{int}; \quad \overbrace{int*}$$