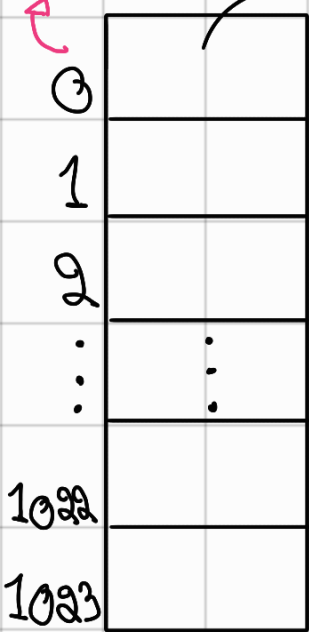


Ponteiros

endereços de memória



Pode ser de 1 byte
8 BITS

TOTAL: 1 KiB

Um **ponteiro** é uma variável que armazena **endereços de memória**.

$$\begin{aligned} & \vdots \\ 1 \text{ KiB} &= 2^{10} \text{ B} \\ 1 \text{ GiB} &= 2^{10} \text{ KiB} = 2^{20} \text{ B} \\ 1 \text{ TiB} &= 2^{30} \text{ B} \\ & \vdots \end{aligned}$$

Classificação da memória principal

Variáveis

Variáveis são porções da memória para armazenar dados.

- Possuem **tipo** para distinguir **tamanho** ocupado na memória e forma de endereçamento.

↳ Principais tipos primitivos:

char - 1 B	SHORT - 2 B
int - 4 B	Float - 4 B
LONG - 8 B	DOUBLE - 8 B

Use o operador **sizeof** para descobrir o tamanho de um tipo ou variável.

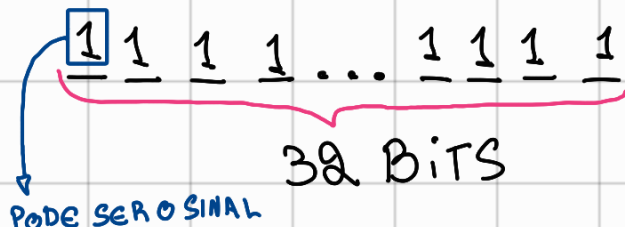
Ex.:

```
INT x;
```

```
PRINTF("%d", sizeof(x)); //4
```

```
PRINTF("%d", sizeof(DOUBLE)); //8
```

Obs.: Um inteiro de 4 B:



$$\underbrace{1 \ 1 \ 1 \ 1 \dots 1 \ 1 \ 1 \ 1}_{32 \text{ BITS}} = 2^0 + 2^1 + 2^2 + \dots + 2^{31}$$
$$= \frac{1(2^{32} - 1)}{2 - 1} = 2^{32} - 1$$
$$= 4.294.967.295$$

este é o limite de um **unsigned int**.

Para o **inteiro** : -2^{31} a $2^{31} - 1$.

Para o **FLOAT** : $\pm 10^{-38}$ a $\pm 10^{38}$

Para o **DOUBLE** : $\pm 10^{-308}$ a 10^{308}

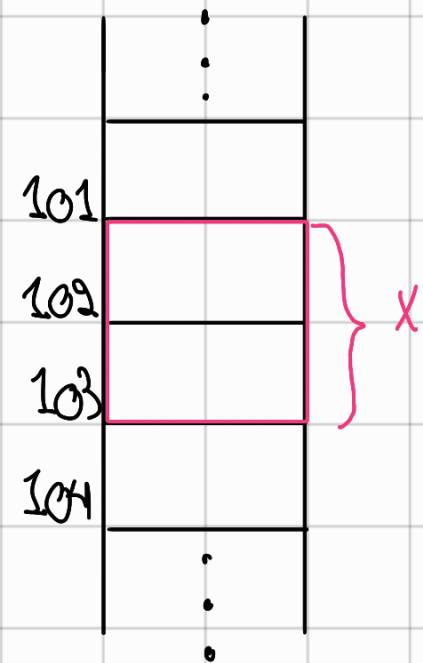
- Possuem endereço de memória.

Quando declaramos uma variável, e.g.:

INT X;

acontece a **alocação automática**:

- ① O S.O. encontra um pedaço de memória disponível
 - ② Reserva o pedaço para a variável.
- O tamanho do pedaço é determinado pelo **tipo** no ex:



Dizemos que:

① endereço de X é 102.