STRINGHE = ARRAY char

```
char s[100];
s[0] = 'c';
s[1] = 'i';
s[2] = 'a';
s[3] = 'o';
s[4] = '\0';
```

'o'

EOS

$\leftrightarrow$ `s[4] = 0;`

```
printf("%s\n", s);
```

```
#include <string.h>
strcpy(s, t);
```

```
strcpy(s, "Ciao");
```

PUNTATORI

C

```
int *p;
int x;
p = &x;
*p = 17;
```

GO

```
p  *int
x  int
p = &x
*p = 17
```

NEW HEAP

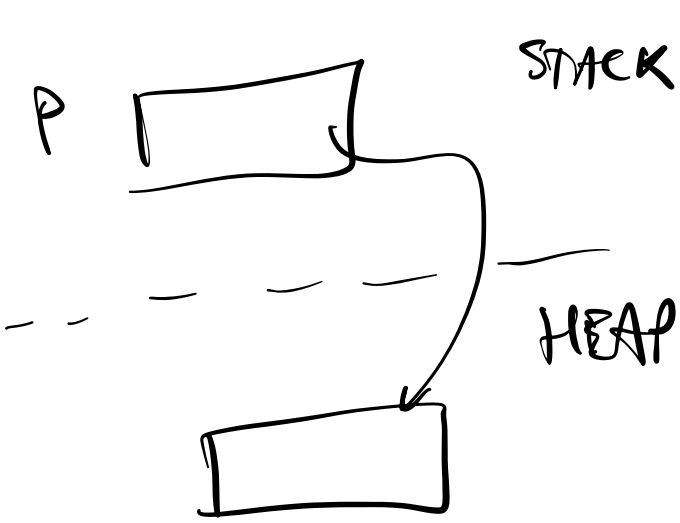
C

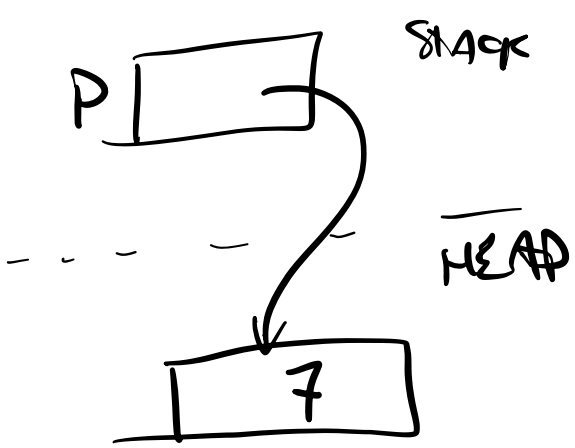
```
int *p;
p=(int *)malloc(sizeof(int));
*p=7;
free (p);
```

(<stdlib.h>)

GO

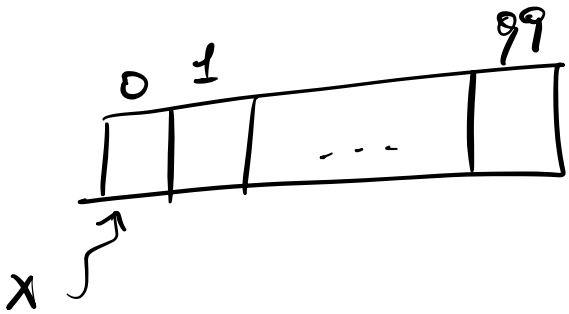
```
p  *int
p= new (int)
*p = 7
```





PUNTATORI E ARRAY

```
int x[100];
```



III. indirizzo di inizio dell'array

$x \equiv \&(x[0])$

```
void fill_zero(int *a, int n) {
    int i;
    for (i=0; i<n; i++)
        a[i]=0

}
int main() {
    int x[20];
    fill_zero(x, 20);
    for (i=0; i<20; i++)
        printf("%d\n", (x[i]+1)*(x[i]+1));

}
```

```
char s[100],*p;
p=s;
strcpy(s, "Paolo Boldi");
s[2] = 'b';
printf("%s\n", s)
```

```
char *s;
s = (char *)malloc(100*sizeof(char));

strcpy(s, "Paolo Boldi");
s[2] = 'b';
printf("%s\n", s)
```

ARITMETICA DEI PUNTATORI

puntatore + int

```
int s[100], *p, *q;

p = s + 2;

*p = 7;
```

p = &(s[2])

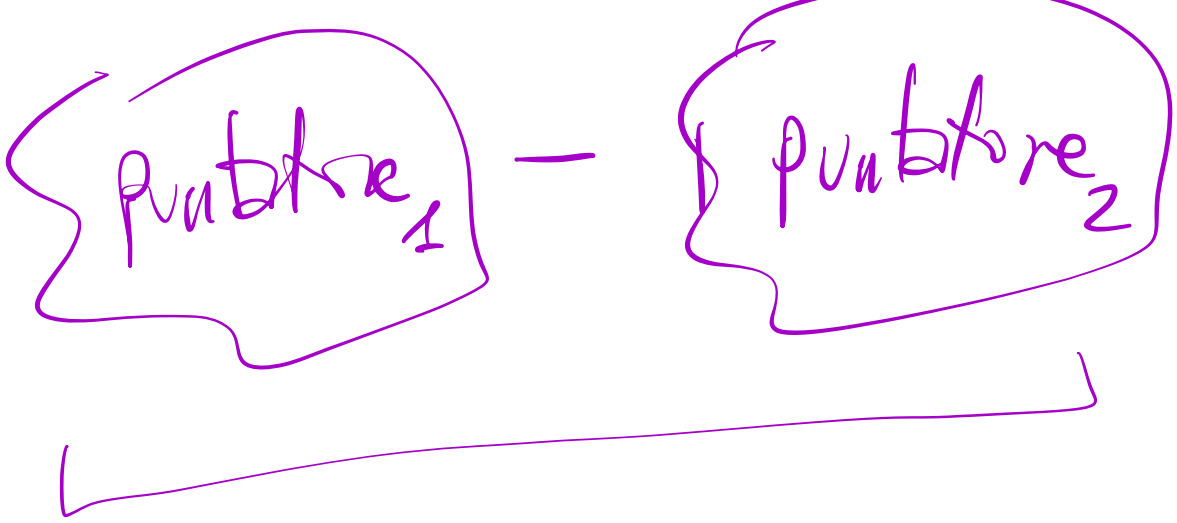
s[2] = 7;

s[2] = ...

*(s+2) = ...

```
char x[100];
strcpy(x, "Paolo Baldini");
char *p;
for (p=x; *p; p++)
    printf("%c", *p);
```

```
for (i=0; p[i]!='\0'; i++)
    printf("%c", p[i]);
```


puntatore1
puntatore2