



POLITECNICO
DI MILANO

INFORMATICA

Successivi
ampliamenti del
dominio di un
programma

```
//Programma che pospone il nome al cognome con ciclo for
```

```
#include <iostream.h>
#include <iomanip.h>
```

```
char carattere; // variabile globale
const char spazio = ' ';
```

```
void saltaspazi() // dichiarazione di funzione
{
    do
    { cin >> carattere;
      } while (carattere == spazio);
}
```

```
void main()
{
    const char tappo = '.';
    int i, j;
    const int lung_max = 20;
    char nome[lung_max];

    cin.unsetf(ios::skipws); //evita in lettura il salto degli spazi

    // inserimento di nome e cognome
    cout << "Inserire nome e cognome separati da uno o più spazi"
          << endl << " con al termine il carattere " << tappo << endl;
    // salta spazi iniziali
    saltaspazi();
    // conserva il nome
```

```
//Programma che pospone il nome al cognome con ciclo for
```

```
#include <iostream.h>
```

```
#include <iomanip.h>
```

```
char carattere;
```

```
// variabile globale
```

```
const char spazio = ' ';
```

```
void saltaspazi() // dichiarazione di funzione
```

P	i	e	r	l	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do  
{ cin >> carattere;  
  } while (carattere == spazio);  
}
```

```
void main()
```

```
{  
  const char tappo = '.';  
  int i, j;  
  const int lung_max = 20;
```



Della Vigna Pierluigi

```
// conserva il nome
i = 0;
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// salta spazi tra nome e cognome
saltaspazi();
// leggi e scrivi i caratteri del cognome
do
{ cout << carattere;
  cin >> carattere;
} while (carattere != tappo);
// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
}
}
```

```
// conserva il nome
i = 0;
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// salta spazi tra nome e cognome
saltaspazi():
do
{ cout << carattere;
  cin >> carattere;
} while (carattere != tappo);
// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
}
```

P	i	e	r	l	u	i	g	i				D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

Della Vigna Pierluigi

```
// conserva il nome
i = 0;
do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio);
// salta spazi tra nome e cognome
saltaspazi();
// leggi e scrivi i caratteri del cognome
do
    { cout << carattere;
      i = i + 1;
    } while (carattere != '\0');
// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
    { cout << nome[j];
      }
}
```

successioni continue di caratteri

```
// conserva il nome
i = 0;
do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio);
// salta spazi tra nome e cognome
saltaspazi();
// leggi e scrivi i caratteri del cognome
do
    { cout << carattere;
      cin >> carattere;
    } while (carattere != tappo);
// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
    { cout << nome[j];
      }
}
```



```
// conserva il nome
    i = 0;
    do
        { nome[i] = carattere;
          i++;
          cin >> carattere;
        } while (carattere != spazio);
// salta spazi tra nome e cognome
    saltaspazi();
// leggi e scrivi i caratteri del cognome
    do
        { cout << carattere;
          cin >> carattere;
        } while (carattere != tappo);
// scrivi i caratteri del nome
    cout << spazio;
    for (j = 0; j <= i-1; j++);
        { cout << nome[j];
          }
}
```

co gno me

Segundo de Sarrión y Díaz de Herrera


```
// conserva il nome
i = 0;
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// salta spazi tra nome e cognome
saltaspazi();
// leggi e scrivi i caratteri del cognome
do
{ cout << carattere;
  cin >> carattere;
} while (carattere != tappo);
// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
}
}
```

P	i	e	r	l	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
// conserva il nome
i = 0;
do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio);
// salta spazi tra nome e cognome
saltaspazi();
// leggi e scrivi i caratteri del cognome
do
    { cout << carattere;
      cin >> carattere;
    } while (carattere != tappo);
// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
    { cout << nome[j];
    }
}
```

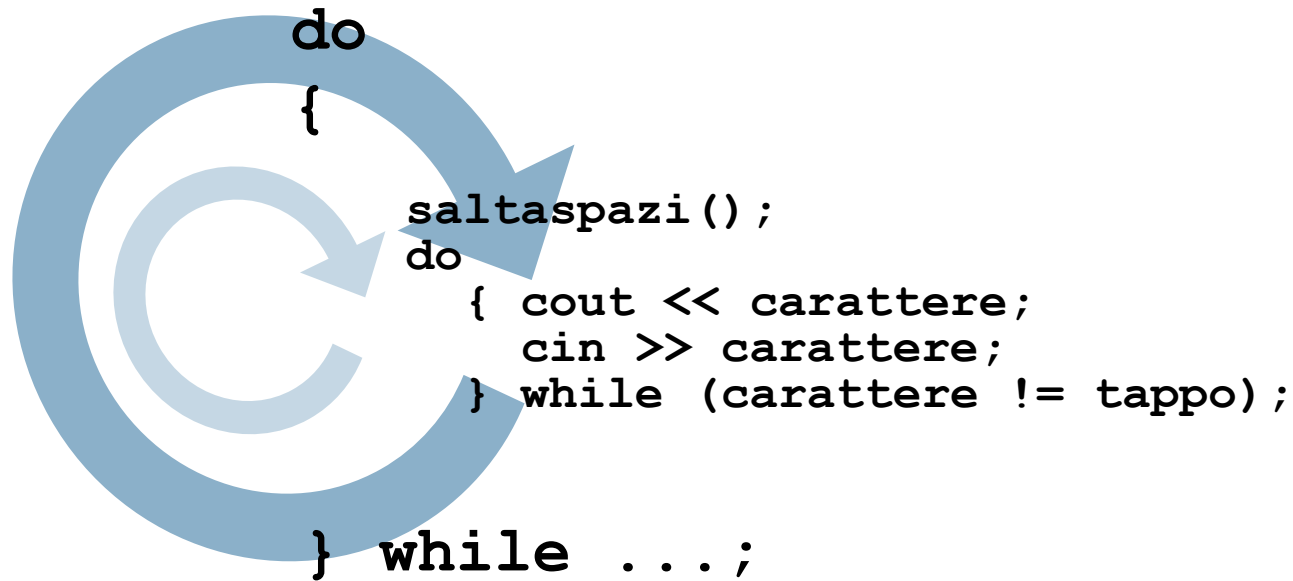


Rossi

Rossi

```
// conserva il nome
i = 0;
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// salta spazi tra nome e cognome
saltaspazi();
// leggi e scrivi i caratteri del cognome
do
{ cout << carattere;
  cin >> carattere;
} while (carattere != tappo);
// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
}
}
```

P	i	e	r	l	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---



P	i	e	r	l	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do  
{
```

```
    saltaspazi();
```

```
    do
```

```
        { cout << carattere;
```

```
          cin >> carattere;
```

```
        } while (carattere != tappo);
```

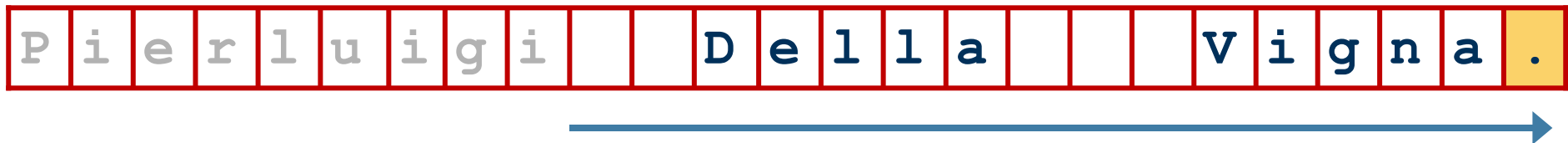
*parte del
cognome*

```
} while ?;
```

P	i	e	r	l	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

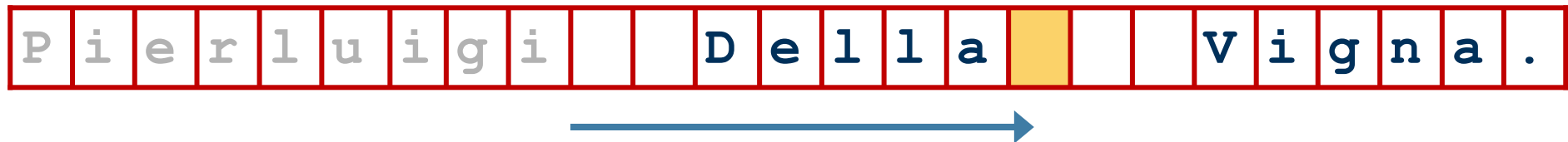
```
do
{
    saltaspazi();
    do
    { cout << carattere;
      cin >> carattere;
    } while (carattere != tappo);

} while ?;
```



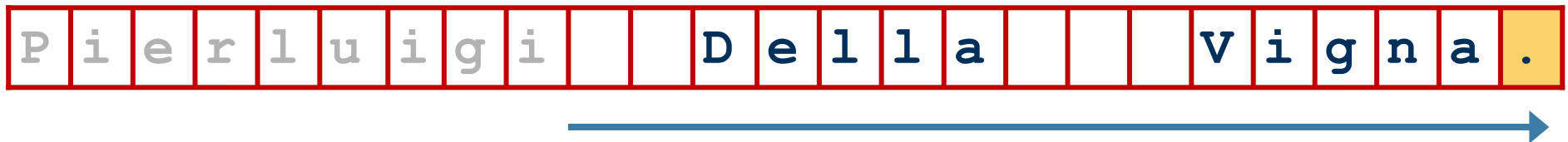
```
do
{
    saltaspazi();
    do
    { cout << carattere;
      cin >> carattere;
    } while (carattere != tappo);

} while ?;
```



```
do
{
    saltaspazi();
    do
    { cout << carattere;
      cin >> carattere;
    } while (carattere != tappo);

} while (carattere != tappo);
```




```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspaзи();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
  }
}
```


```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
  }
}
```

P	i	e	r	l	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
  }
}
```



P	i	e	r	l	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
  }
}
```

P	i	e	r	l	u	i	g	i			D	e	l	l	a			V	i	g	n	a	.
---	---	---	---	---	---	---	---	---	--	--	---	---	---	---	---	--	--	---	---	---	---	---	---

```
do
{ nome[i] = carattere;
  i++;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
}
}
```

carattere == tappo || carattere == spazio

$$!(a \ \&\& \ b) = (!a \ || \ !b)$$

```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
  }
}
```

P	i	e	r		L	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
}
```

*nome**cognome*

P	i	e	r		L	u	i	g	i		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
  }
}
```

*nome**cognome*

P	i	e	r		L	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---


```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspaio();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = i-1; j >= 0; j--)
{ cout << nome[j];
  }
}
```

Luca
Elisa
Teresa
Ignazio
Angela
Matteo
Oreste
Piero
Ettore
Rosa

Sironi
Eco
Majorana
Pavolini
Rambaldi
Esposito
Tasso
Ungaretti
Olmi
Mazzali

P	i	e	r		L	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);
// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++)
{ cout << nome[j];
  }
}
```

Angela

Piero Angela

Piero

P	i	e	r		L	u	i	g	i			D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	--	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);
// leggi e scrivi i caratteri del cognome
do
{
  saltaspazi();
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
  }
}
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
cin.unsetf(ios::skipws);      //evita in lettura il salto degli spazi

// inserimento di nome e cognome
cout << "Inserisci nome e cognome separati da uno o più spazi"
    << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspaзи();
// conserva il nome
i = 0;
do
{ nome[i] = carattere;
  i++;
  cin >> carattere;
} while (carattere != spazio);

// leggi e scrivi i caratteri del cognome
do
{
    saltaspaзи();
    do
    { cout << carattere;
      cin >> carattere;
    } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
cout << spazio;
for (j = 0; j <= i-1; j++);
{ cout << nome[j];
  }
}
```

...

```
saltaspazi();
```

```
i = 0
```

```
do
```

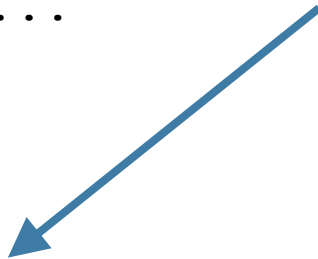
```
{ nome[i] = carattere;
```

```
  i++;
```

```
  cin >> carattere;
```

```
} while (carattere != spazio);
```

...



P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do
{
    saltaspaзи();
    i = 0
do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio);
} while (carattere != tappo);
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
do
{
    saltaspazi();
    i = 0
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
} while (carattere != tappo);
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome
do
{
    saltaspazi();
    i = 0;
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
  } while (carattere != tappo);

// leggi e scrivi i caratteri del cognome
do
```

Della Vigna Luigi


```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome
do
{
    saltaspazi();
    i = 0;
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
  } while (carattere != tappo);

// leggi e scrivi i caratteri del cognome
do
```

Della Vigna Luigi

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome
do
{
    saltaspazi();
    i = 0;
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
  } while (carattere != tappo);

// leggi e scrivi i caratteri del cognome
do
```




A diagram illustrating a character array (string) storage. The array is represented as a horizontal row of 25 cells, each containing a character. The characters are: P, i, e, r, (empty), L, u, i, g, i, ., (empty), D, e, l, l, a, (empty), (empty), (empty), V, i, g, n, a, .. The first four cells (P, i, e, r) are highlighted in red. The fifth cell is highlighted in yellow. A blue arrow points from the yellow cell to the code line `carattere != spazio` in the code block above.

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome
do
{
    saltaspazi();
    i = 0;
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
  } while (carattere != tappo);

// leggi e scrivi i caratteri del cognome
do
```



P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome
i = 0;
do
{
    saltaspazi();
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
  } while (carattere != tappo);

// leggi e scrivi i caratteri del cognome
do
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

Nome

[0] [1] [2] [3]

[4] [5] [6] [7] [8]

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;
```

```
// salta spazi iniziale
saltaspazi();
// conserva il nome
i = 0;
do
{
    saltaspazi();
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
} while (carattere != tappo);
```

```
// leggi e scrivi i caratteri del cognome
do
```



```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspaзи();
// conserva il nome
i = 0;
do
{
    saltaspaзи();
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
    if (carattere == spazio)
    { nome[i] = spazio;
      i++;
    }
} while (carattere != tappo);

// leggi e scrivi i caratteri del cognome
do
{
    saltaspaзи();
    do
    { cout << carattere;
      cin >> carattere;
    } while (carattere != tappo && carattere != spazio);
} while (carattere != tappo);

// scrivi i caratteri del nome
```

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
  saltaspazi();
  // conserva una parte del nome
  do
  { nome[i] = carattere;
    i++;
    cin >> carattere;
  } while (carattere != spazio && carattere != tappo);
  // metti spazio in nome, se non è l'ultima parte
  if (carattere == spazio)
  { nome[i] = spazio;
    i++;
  }
} while (carattere != tappo);

// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo);
}
```

```
const char tappo = '.';
int i, j;
const int lung_max = 20;
char nome[lung_max];
cin.unsetf(ios::skipws);    //evita in lettura il salto degli spazi
// inserimento di nome e cognome
    cout << "Inserisci nome e cognome in successione, ognuno" << endl
         << "immediatamente seguito da " << tappo << " ;" << endl
         << "nome e cognome possono essere composti;" << endl
         << "le parti sono separate da uno o più spazi." << endl;

// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
    saltaspazi();
    // conserva una parte del nome
    do
        { nome[i] = carattere;
          i++;
```

Inserisci nome e cognome in successione, ognuno
immediatamente seguito da .;
nome e cognome possono essere composti;
le parti sono separate da uno o più spazi.


```
const char tappo = '.';
int i, j;
const int lung_max = 20;
char nome[lung_max];
cin.unsetf(ios::skipws);    //evita in lettura il salto degli spazi
// inserimento di nome e cognome
    cout << "Inserisci nome e cognome in successione, ognuno" << endl
         << "immediatamente seguito da " << tappo << " ;" << endl
         << "nome e cognome possono essere composti;" << endl
         << "le parti sono separate da uno o più spazi." << endl;

// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
    saltaspazi();
    // conserva una parte del nome
    do
        { nome[i] = carattere;
          i++;
          cin >> carattere;
        } while (carattere != spazio && carattere != tappo);
    // metti spazio in nome, se non è l'ultima parte
    if (carattere == spazio)
        { nome[i] = spazio;
          i++;
        }
    } while (carattere != tappo);

// leggi e scrivi cognome composto da una o più parti
```

```
const char tappo = '.';
int i, j;
const int lung_max = 20;
char nome[lung_max];
cin.unsetf(ios::skipws);    //evita in lettura il salto degli spazi
// inserimento di nome e cognome
    cout << "Inserisci nome e cognome in successione, ognuno" << endl
         << "immediatamente seguito da " << tappo << " ;" << endl
         << "nome e cognome possono essere composti;" << endl
         << "le parti sono separate da uno o più spazi." << endl;

// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
    saltaspazi();
    // conserva una parte del nome
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
    // metti spazio in nome, se non è l'ultima parte
    if (carattere == spazio)
    { nome[i] = spazio;
      i++;
    }
} while (carattere != tappo);

// leggi e scrivi cognome composto da una o più parti
```

```
const char tappo = '.';
int i, j;
const int lung_max = 20;
char nome[lung_max];
cin.unsetf(ios::skipws);    //evita in lettura il salto degli spazi
// inserimento di nome e cognome
    cout << "Inserisci nome e cognome in successione, ognuno" << endl
         << "immediatamente seguito da " << tappo << " ;" << endl
         << "nome e cognome possono essere composti;" << endl
         << "le parti sono separate da uno o più spazi." << endl;

// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
    saltaspazi();
    // conserva una parte del nome
    do
    { nome[i] = carattere;
      i++;
      cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
    // metti spazio in nome, se non è l'ultima parte
    if (carattere == spazio)
    { nome[i] = spazio;
      i++;
    }
} while (carattere != tappo);

// leggi e scrivi cognome composto da una o più parti
```

```
const char tappo = '.';
int i, j;
const int lung_max = 20;
char nome[lung_max];
cin.unsetf(ios::skipws);    //evita in lettura il salto degli spazi
// inserimento di nome e cognome
    cout << "Inserisci nome e cognome in successione, ognuno" << endl
         << "immediatamente seguito da " << tappo << " ;" << endl
         << "nome e cognome possono essere composti;" << endl
         << "le parti sono separate da uno o più spazi." << endl;

// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
  saltaspazi();
  // conserva una parte del nome
  do
  { nome[i] = carattere;
    i++;
    cin >> carattere;
  } while (carattere != spazio && carattere != tappo);
  // metti spazio in nome, se non è l'ultima parte
  if (carattere == spazio)
  { nome[i] = spazio;
    i++;
  }
} while (carattere != tappo);

// leggi e scrivi cognome composto da una o più parti
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

```
// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
  saltaspazi();
  // conserva una parte del nome
  do
  { nome[i] = carattere;
    i++;
    cin >> carattere;
  } while (carattere != spazio && carattere != tappo);
  // metti spazio in nome, se non è l'ultima parte
  if (carattere == spazio)
  { nome[i] = spazio;
    i++;
  }
} while (carattere != tappo);

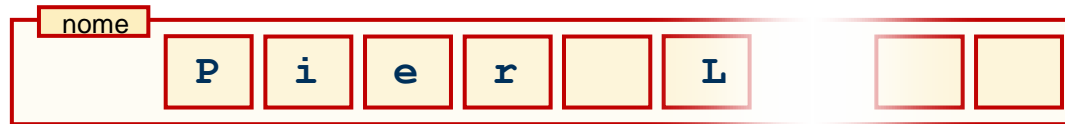
// leggi e scrivi cognome composto da una o piu' parti
```

```
        i++;
        cin >> carattere;
    } while (carattere != spazio && carattere != tappo);
    // metti spazio in nome, se non è l'ultima parte
    if (carattere == spazio)
    { nome[i] = spazio;
      i++;
    }
} while (carattere != tappo);

// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
  // scrivi spazio
  cout << spazio;
} while (carattere != tappo);

// scrivi i caratteri del nome
for (j = 0; j <= 1; j++)
{ cout << nome[j];
  }
}
```

```
// leggi e scrivi cognome composto da una o piu' parti  
do  
{ // salta spazi prima di parte del cognome  
  saltaspazi();
```



```
    } while (carattere != tappo da carattere != spazio);  
    // scrivi spazio  
    cout << spazio;  
  } while (carattere != tappo);  
  
  // scrivi i caratteri del nome  
  for (j = 0; j <= 1; j++)  
  { cout << nome[j];  
  }  
}
```

Della Vigna

```
// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
  // scrivi spazio
  cout << spazio;
} while (carattere != tappo);

// scrivi i caratteri del nome
for (j = 0; j <= 1; j++)
{ cout << nome[j];
}
}
```

Della Vigna


```
// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
  // scrivi spazio
  cout << spazio;
} while (carattere != tappo);

// scrivi i caratteri del nome
for (j = 0; j <= 1; j++)
{ cout << nome[j];
}
}
```

Della Vigna Pier Luigi

Variante

nome []



Variante

array1[]

array2[]

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

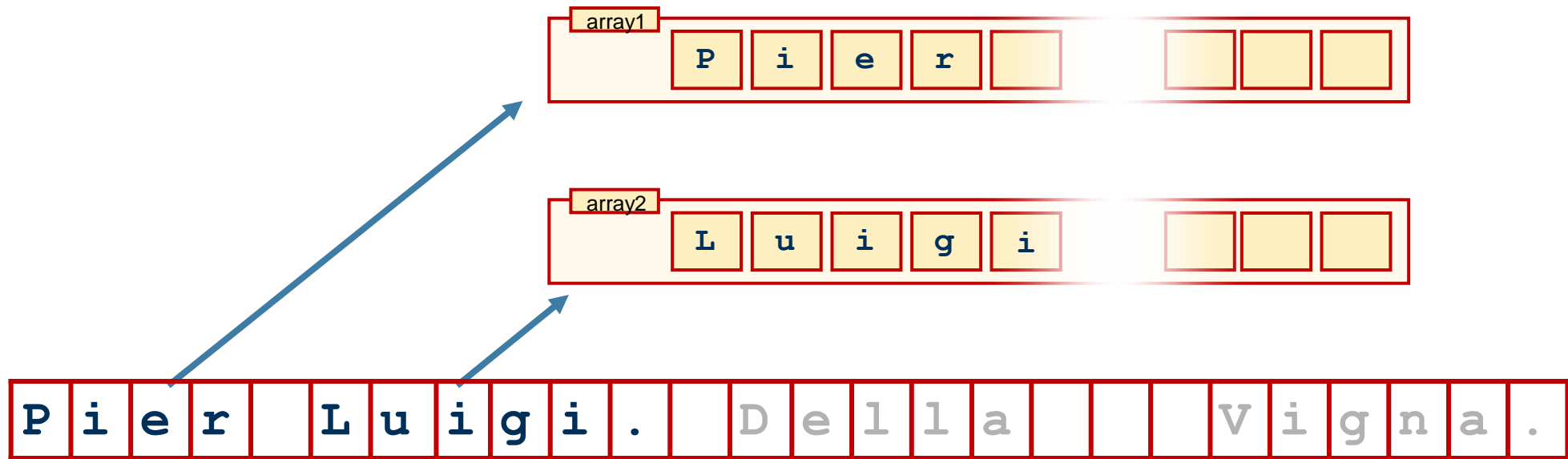
```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;
```

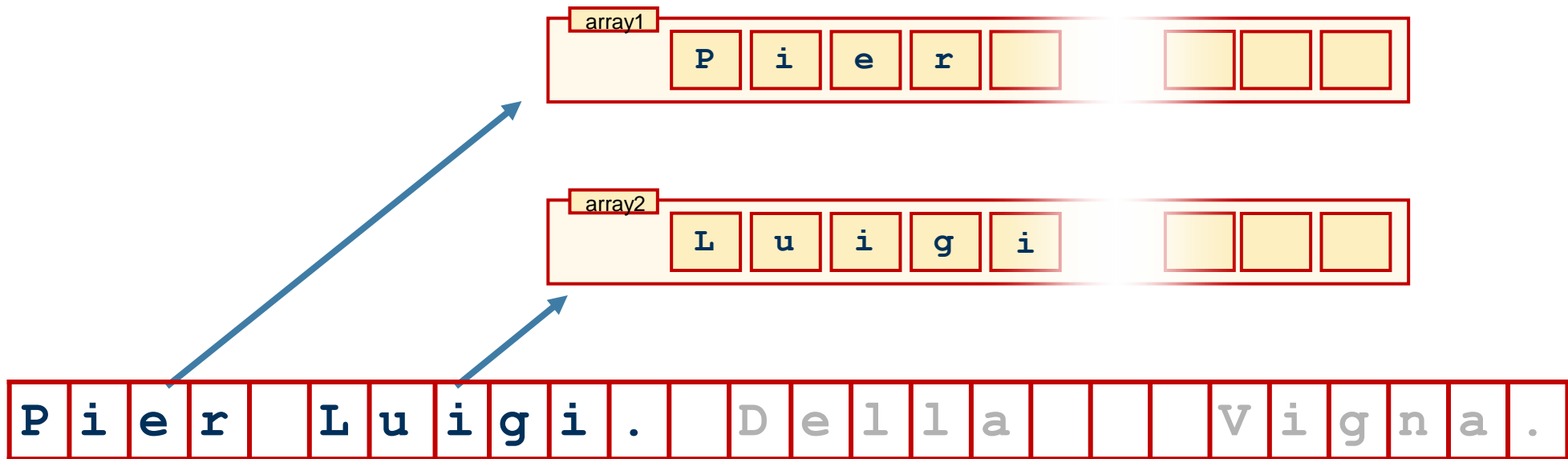
```
// salta spazi iniziale
saltaspazi();
// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
  saltaspazi();
  // conserva una parte del nome
  array1[] do array2[]
```

P i e r . L u i g i . D e l l a V i g n a .

```
if (carattere == spazio)
{ nome[i] = spazio;
  i++;
}
} while (carattere != tappo);
```

```
// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere != tappo);
```





```
// conserva il nome composto
i = 0;
do
{ // salta spazi iniziali
  saltaspazi();
  // conserva una parte del nome
  do
  { nome[i] = carattere;
    i++;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
  // metti spazio in nome, se non è l'ultima parte
  if (carattere == spazio)
  { nome[i] = spazio;
    i++;
```



nome

→	P	i	e	r							
→	L	u	i	g	i						
→	F	i	l	i	p	p	o				
→											

nome

[0] [1] [2] [3] ...

[18] [19]



nome

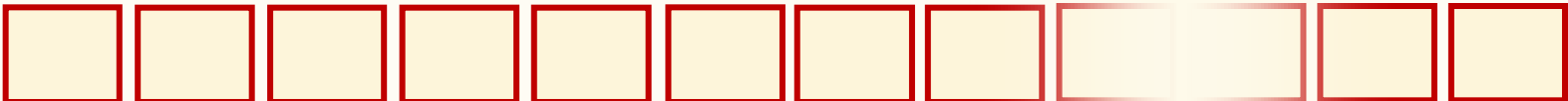
[0]



[1]



[2]



[3]



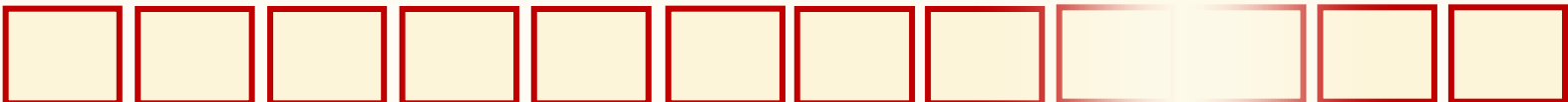
[4]

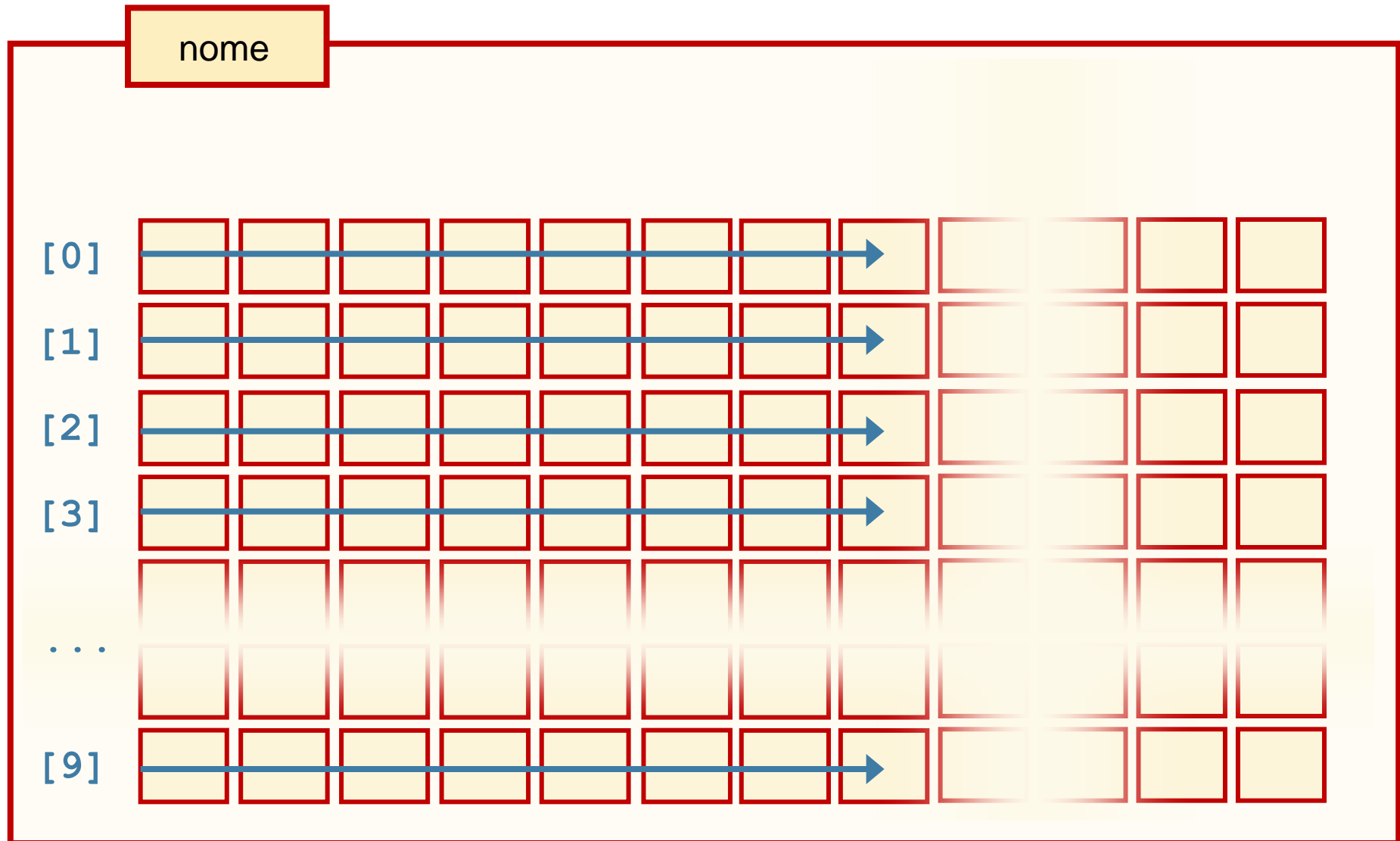


...



[9]





nome

	[0]	[1]	[2]	[3]	...							[18]	[19]
[0]	P	i	e	r									
[1]													
[2]													
[3]													
...													
[9]													

Diagram illustrating memory layout for a string named "nome". The string is stored in a 2D array structure. The first row (index [0]) contains the characters 'P', 'i', 'e', 'r', followed by empty cells. The second row (index [1]) contains empty cells. The third row (index [2]) contains the character 'c' in the first column, followed by empty cells. The word "char" is highlighted in the cell at row [2], column [2]. Arrows point from the characters 'P', 'i', 'e', 'r' in the first row to the 'c' in the third row, indicating a shift or transformation of the string.

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome composto
i = 0;
do
...   salta spazi iniziali
const int numParti = 10;
...   { nome[i] = carattere;
char nome[...][...];
...   } while (carattere != spazio && carattere != tappo);
// metti spazio in nome, se non è l'ultima parte
    if (carattere == spazio)
        { nome[i] = spazio;
          i++;
        }
    } while (carattere != tappo);

// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
      { cout << carattere;
        cin >> carattere;
      } while (carattere !=
```

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome composto
i = 0;
do
...   salta spazi iniziali
const int numParti = 10;
...   { nome[i] = carattere;
char nome [numParti][..];
...   } while (carattere != spazio && carattere != tappo);
      metti spazio in nome, se non è l'ultima parte
      if (carattere == spazio)
      { nome[i] = spazio;
        i++;
      }
    } while (carattere != tappo);

// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere !=
```

```
cout << "Inserisci nome e cognome separati da uno o più spazi"
      << endl << " con al termine il carattere " << tappo << endl;

// salta spazi iniziale
saltaspazi();
// conserva il nome composto
i = 0;

... // salta spazi iniziali
saltaspazi();
const int numParti = 10;
const int lung_max = 20;
char nome [numParti][lung_max];
... // leggi e scrivi il carattere,
... } while (carattere != spazio && carattere != tappo);
// tutti spazio in nome, se non è l'ultima parte
if (carattere == spazio)
{ nome[i] = spazio;
  i++;
}
} while (carattere != tappo);

// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
  } while (carattere !=
```

```
void main()
{
    const char tappo = '.';
    int i, j;
    const int numParti = 10;
    const int lung_max = 20;
    char nome[numParti][lung_max];
    cin.unsetf(ios::skipws); //evita in lettura il salto degli spazi
    // inserimento di nome e cognome
    cout << "Inserisci nome e cognome in successione, " << endl
         << "ognuno immediatamente seguito da " << tappo << endl
         << "Nome e cognome possono essere composti." << endl
         << "Le parti sono separate da uno o più spazi." << endl;

    // conserva il nome composto
    i = 0;
    do
    { // salta spazi iniziali
        saltaspazi();
        // conserva una parte del nome
        do
        { nome[i] = carattere;
          i++;
          cin >> carattere;
        } while (carattere != spazio && carattere != tappo);
        // metti spazio in nome, se non è l'ultima parte
        if (carattere == spazio)
        { nome[i] = spazio;
          i++;
        }
    } while (carattere != tappo);
```



```
<< "Le parti sono separate da uno o più spazi. << endl;
```

```
// conserva il nome composto
```

```
i = 0;
```

```
do
```

```
{ // salta spazi iniziali
```

```
  saltaspaзи();
```

```
  // conserva una parte del nome
```

```
  do
```

```
    { nome[i] = carattere;
```

nome[?][?]

```
      i++;
```

```
      cin >> carattere;
```

```
    } while (carattere != tappo && carattere != spazio);
```

```
  // metti spazio in nome, se non è l'ultima parte
```

```
  if (carattere == spazio)
```

```
    { nome[i] = spazio;
```

```
      i++;
```

```
    }
```

```
  } while (carattere != tappo);
```

```
// leggi e scrivi cognome composto da una o piu' parti
```

```
do
```

```
{ // salta spazi prima di parte del cognome
```

```
  saltaspaзи();
```

```
  // leggi e scrivi parte del cognome
```

```
  do
```

```
    { cout << carattere;
```

```
      cin >> carattere;
```

```
    } while (carattere != tappo && carattere != spazio);
```

```
  // scrivi spazio
```

```
  cout << spazio;
```

```
} while (carattere != tappo);
```

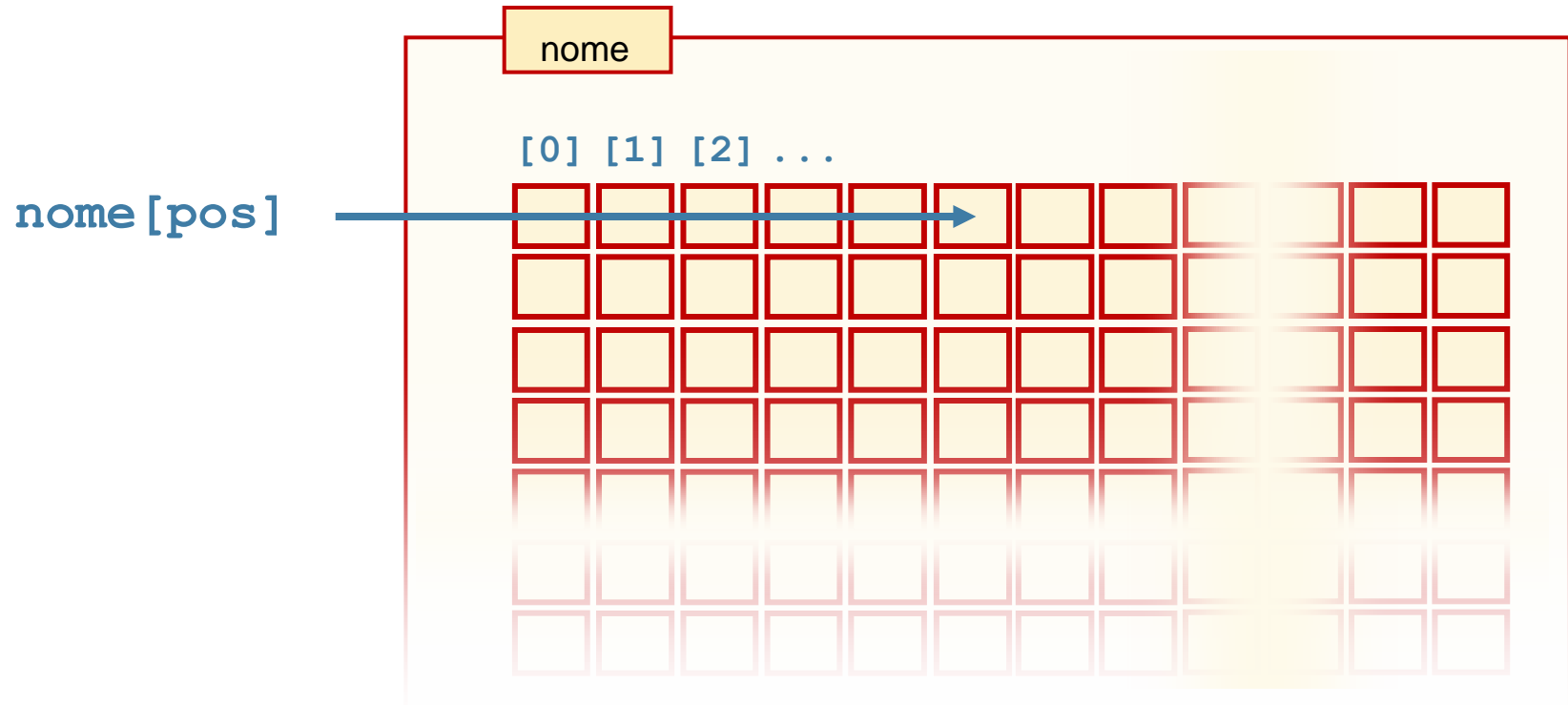
```
...  
i = 0;  
do  
{ saltaspazi();  
  do  
  { nome[i] = carattere;  
    i++;  
    cin >> carattere;  
  } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
  { nome[i] = spazio;  
    i++;  
  }  
} while (carattere != tappo);  
...
```

```
...  
i = 0;  
do  
{ saltaspazi();  
  do  
  { nome[i] = carattere;  
    i++;  
    cin >> carattere;  
  } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
  { nome[i] = spazio;  
    i++;  
  }  
} while (carattere != tappo);  
...
```

```
...  
i = 0;  
do  
{ saltaspazi();  
  do  
  { nome[pos][i] = carattere;  
    i++;  
    cin >> carattere;  
  } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
  { nome[i] = spazio;  
    i++;  
  }  
} while (carattere != tappo);  
...
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---


nome[pos][]



P	i	e	r		L	u	i	g	i	.		D	e	l	l	a			V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	---	---	---	---	---	---

nome[pos][]

```
...  
i = 0;  
do  
{ saltaspazi();  
  do  
  { nome[pos][i] = carattere;  
    i++;  
    cin >> carattere;  
  } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
  { nome[i] = spazio;  
    i++;  
  }  
} while (carattere != tappo);  
...
```

```
i = 0;
```

~~[i]~~

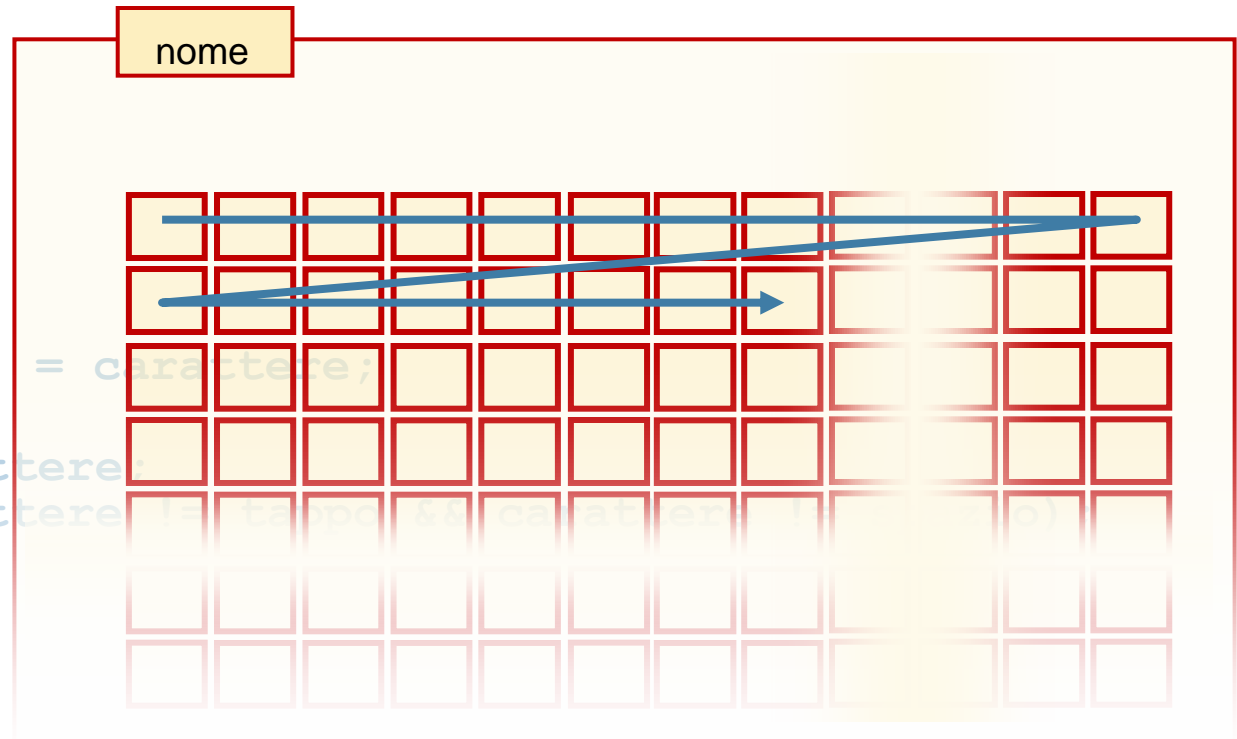
```
do
```

```
{ nome[pos][i] = carattere;
```

```
  i++;
```

```
  cin >> carattere;
```

```
} while (carattere != '\0' && carattere != '\n');
```



```
i = 0;
```

```
nome[pos]
```

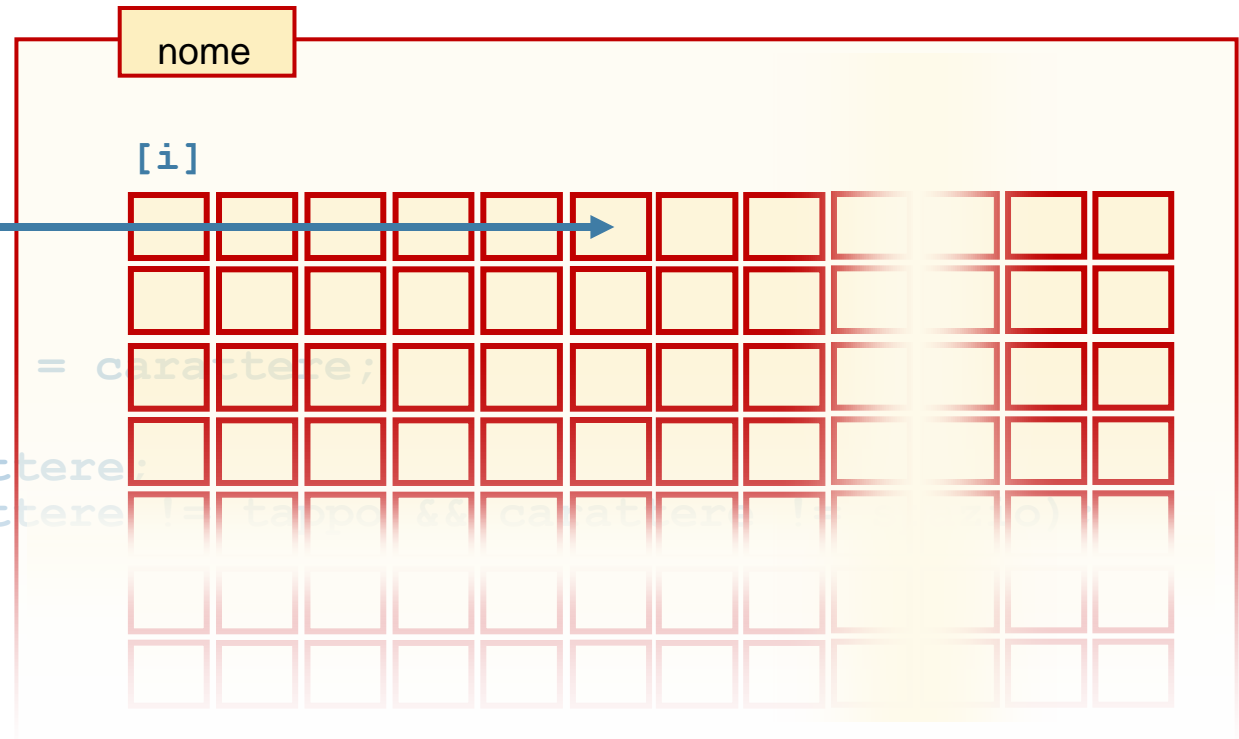
```
do
```

```
{ nome[pos][i] = carattere;
```

```
  i++;
```

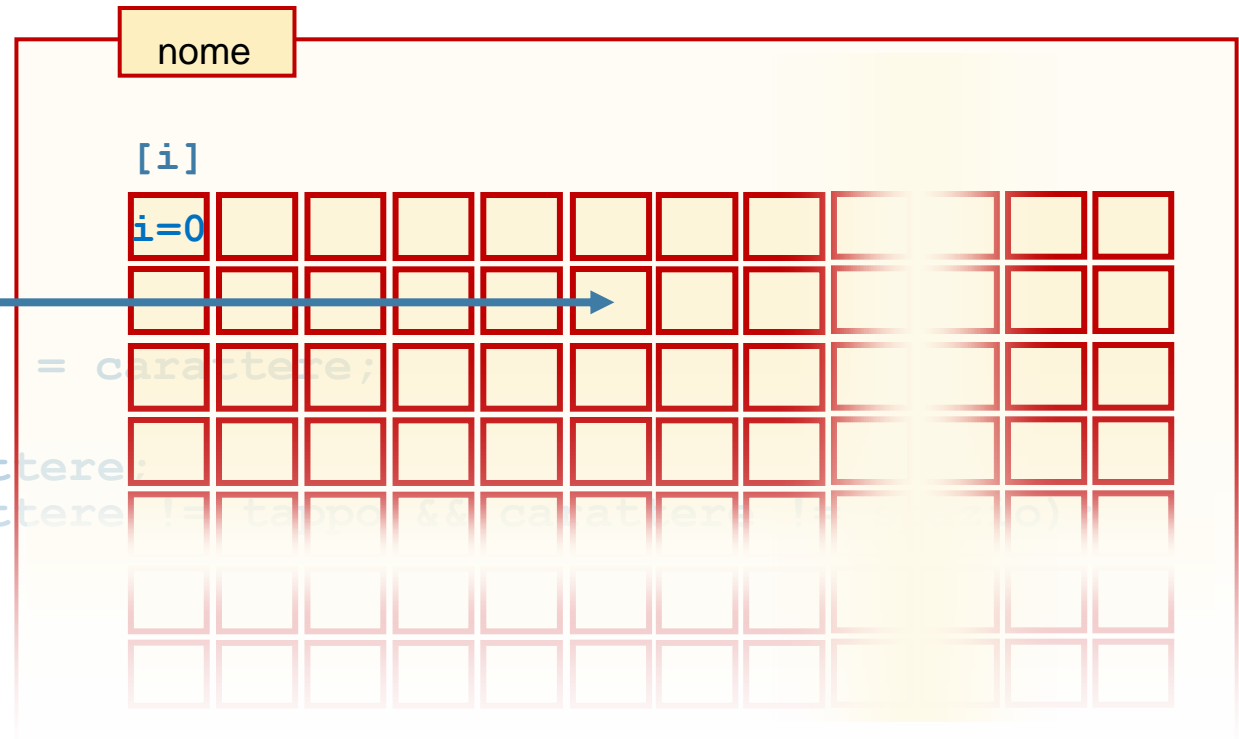
```
  cin >> carattere;
```

```
} while (carattere != '\0' && carattere != '\n');
```




```
i = 0;
```

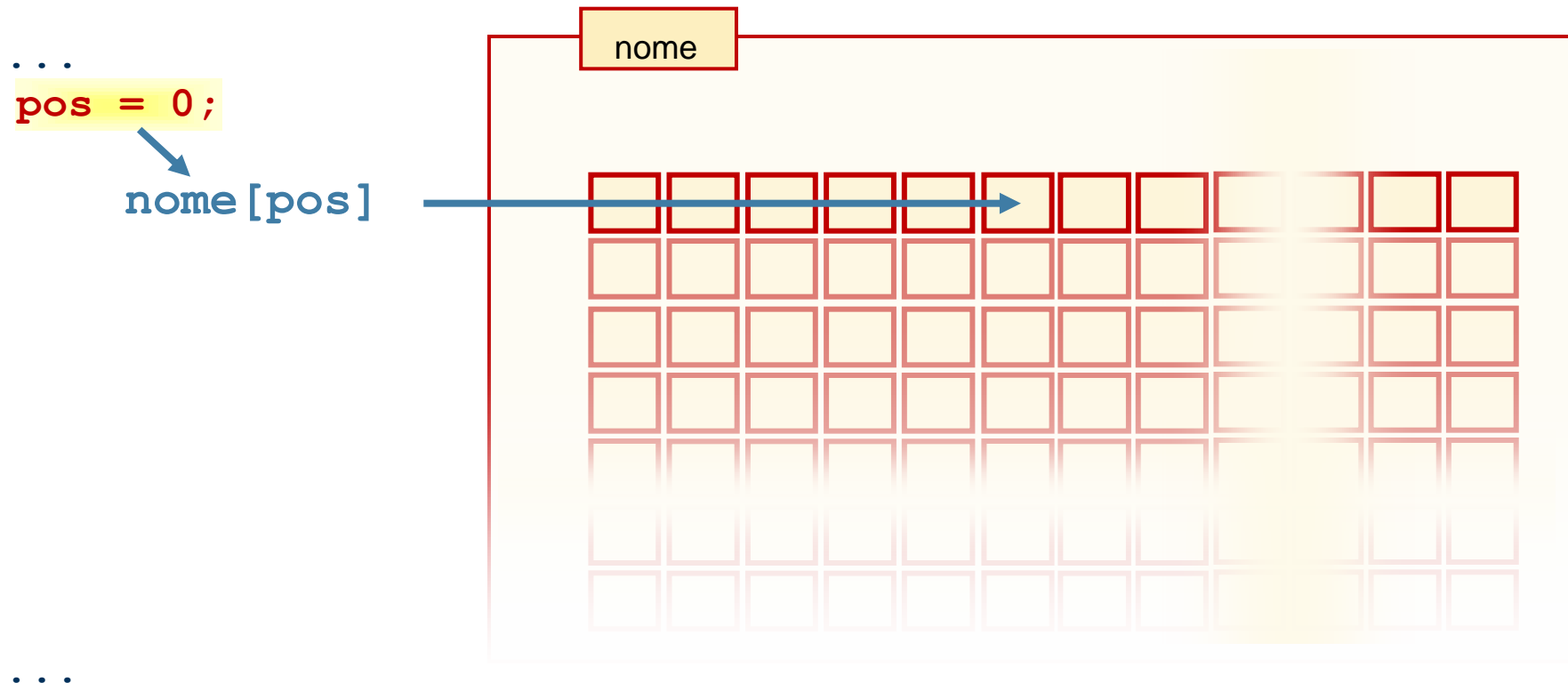
```
do nome[pos]  
{ nome[pos][i] = carattere;  
  i++;  
  cin >> carattere;  
} while (carattere != '\0' && carattere != '\n');
```



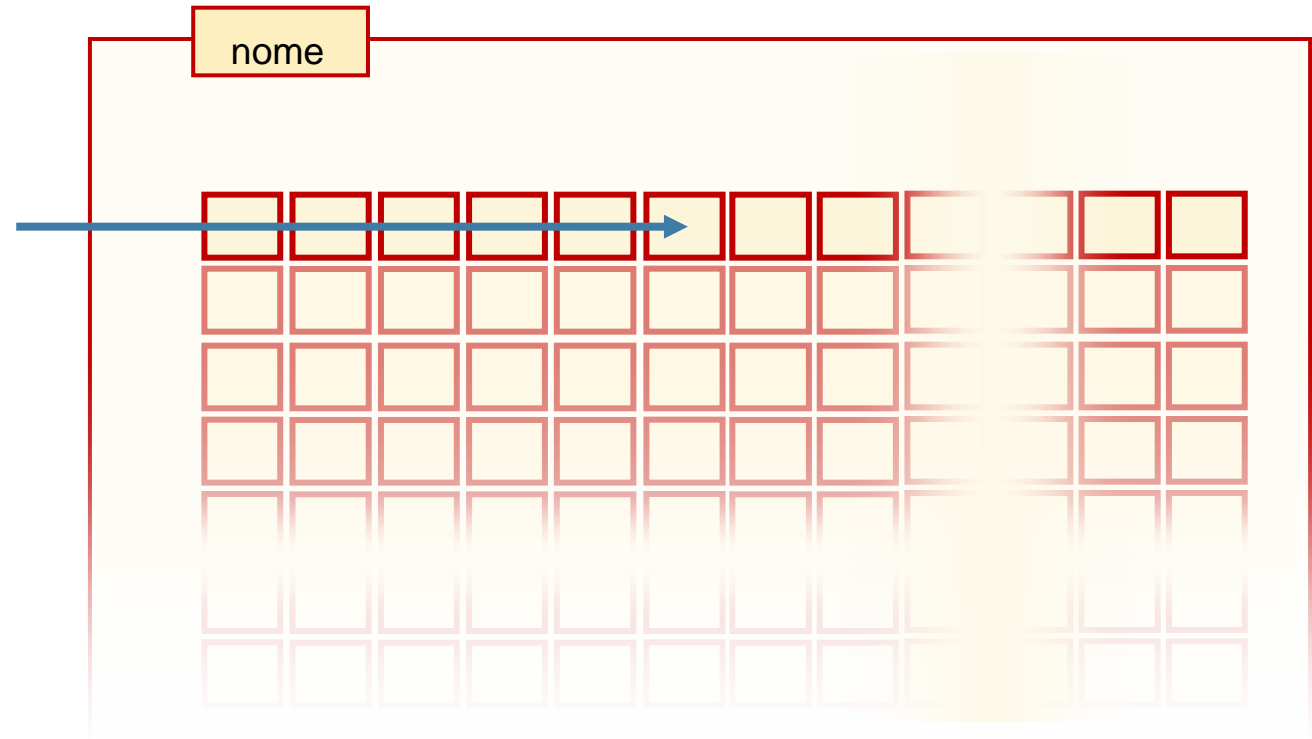
```
...  
i = 0;  
do  
{ saltaspazi();  
  do  
  { nome[pos][i] = carattere;  
    i++;  
    cin >> carattere;  
  } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
  { nome[i] = spazio;  
    i++;  
  }  
} while (carattere != tappo);  
...
```

```
...  
do  
{ saltaspazi();  
  i = 0;  
  do  
  { nome[pos][i] = carattere;  
    i++;  
    cin >> carattere;  
  } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
  { nome[i] = spazio;  
    i++;  
  }  
} while (carattere != tappo);  
...
```

```
...  
pos = 0;  
do  
{ saltaspaзи();  
  i = 0;  
  do  
  { nome[pos][i] = carattere;  
    i++;  
    cin >> carattere;  
  } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
  { nome[i] = spazio;  
    i++;  
  }  
} while (carattere != tappo);  
...
```



...

`pos = 0;``nome[0]`

...

```
...  
pos = 0;  
do  
{ saltaspaзи();  
  i = 0;  
  do  
    { nome[pos][i] = carattere;  
      i++;  
      cin >> carattere;  
    } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
    { nome[i] = spazio;  
      i++;  
    }  
} while (carattere != tappo);  
...
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

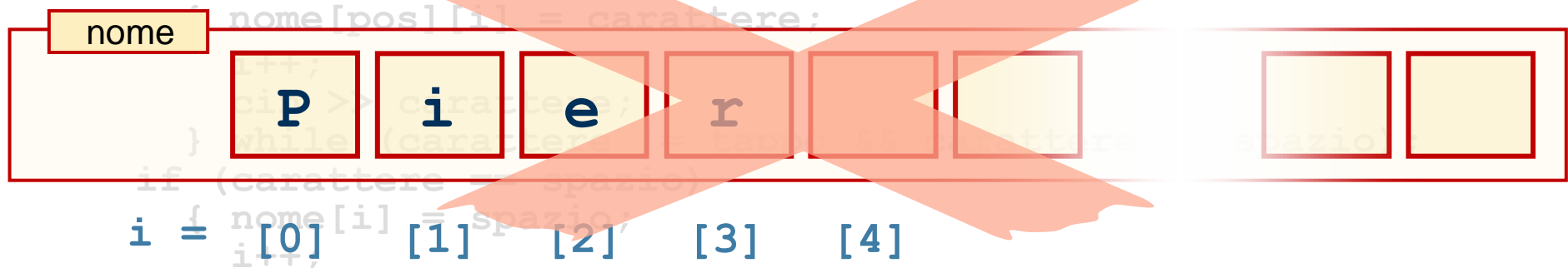
nome[i]

```
...  
pos = 0;  
do  
{ saltaspaзи();  
  i = 0;  
  do  
    { nome[pos][i] = carattere;  
      i++;  
      cin >> carattere;  
    } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
    { nome[i] = spazio;  
      i++;  
    }  
} while (carattere != tappo);  
...
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---

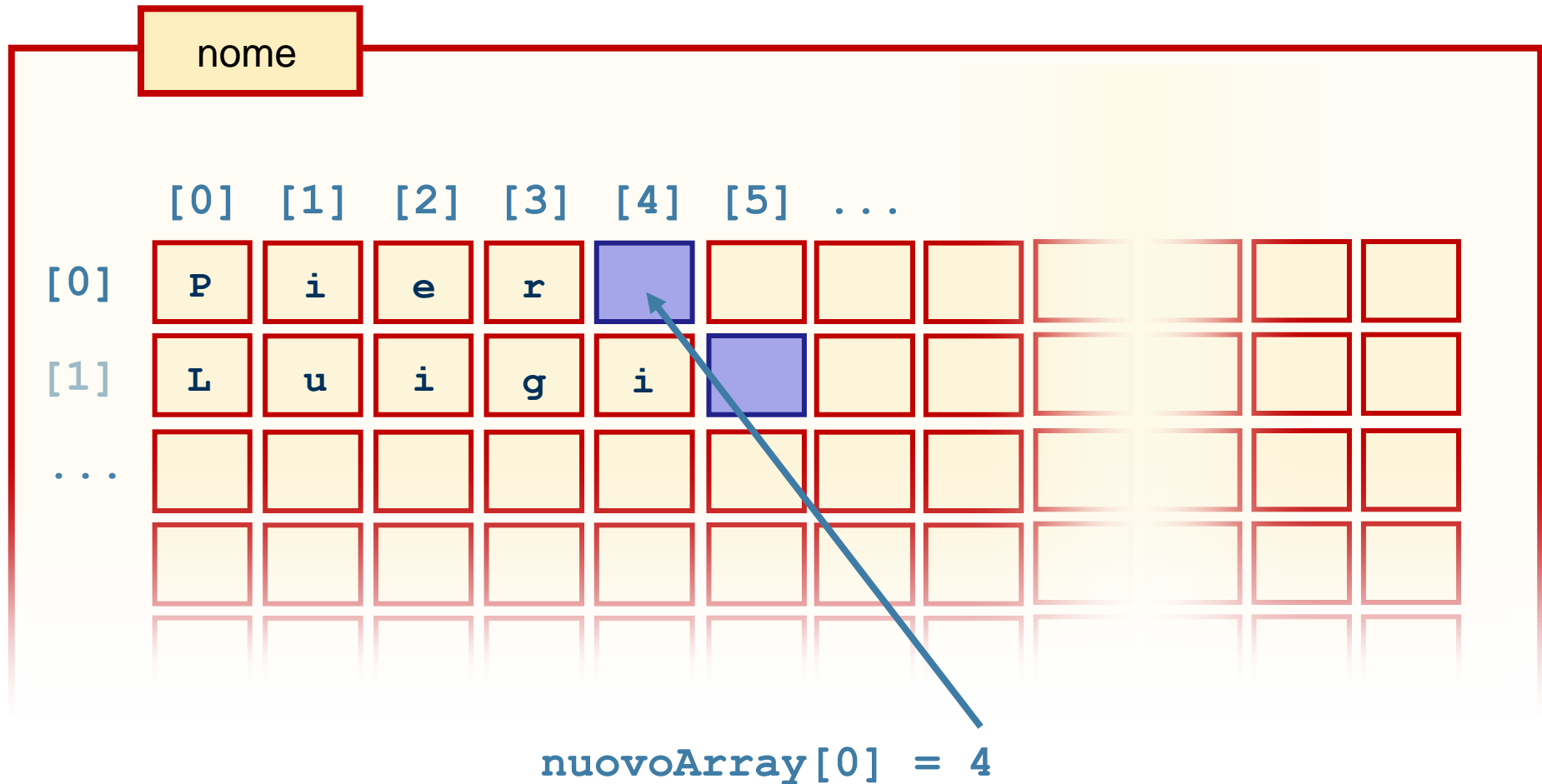
nome[i]


```
...  
pos = 0;  
do  
{ saltaspaзи();  
  i = 0;  
  do
```

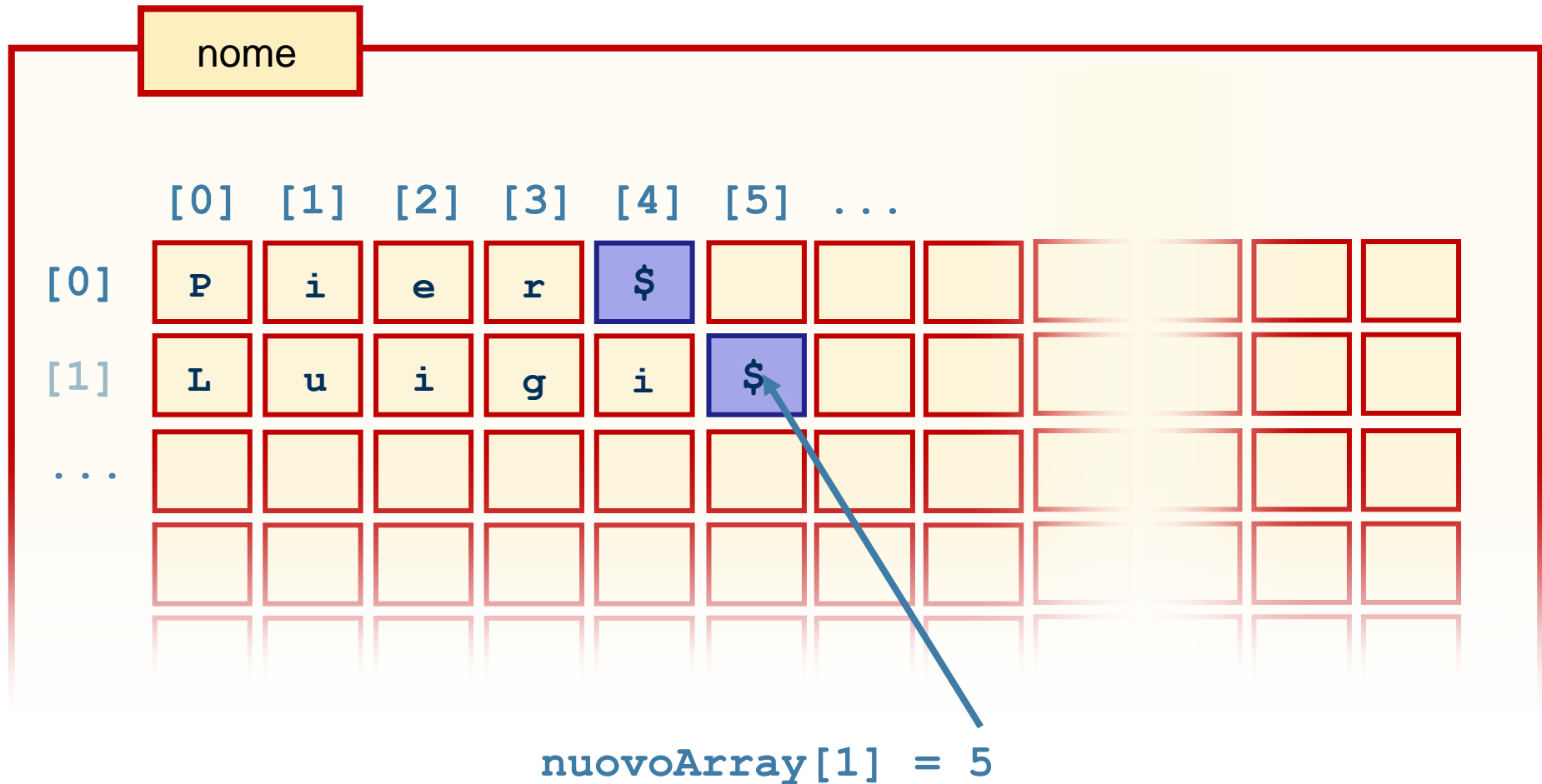


```
  if (carattere == spazio)  
  { nome[i] = spazio;  
    i++;  
  }  
} while (carattere != tappo);  
...
```

P	i	e	r		L	u	i	g	i	.		D	e	l	l	a				V	i	g	n	a	.
---	---	---	---	--	---	---	---	---	---	---	--	---	---	---	---	---	--	--	--	---	---	---	---	---	---



```
int nuovoArray[10];
```



Tappo

```
int nuovoArray[10];
```

```
...  
pos = 0;  
do  
{ saltaspaзи();  
  i = 0;  
  do  
    { nome[pos][i] = carattere;  
      i++;  
      cin >> carattere;  
    } while (carattere != tappo && carattere != spazio);  
  if (carattere == spazio)  
    { nome[i] = spazio;  
      i++;  
    }  
} while (carattere != tappo);  
...
```

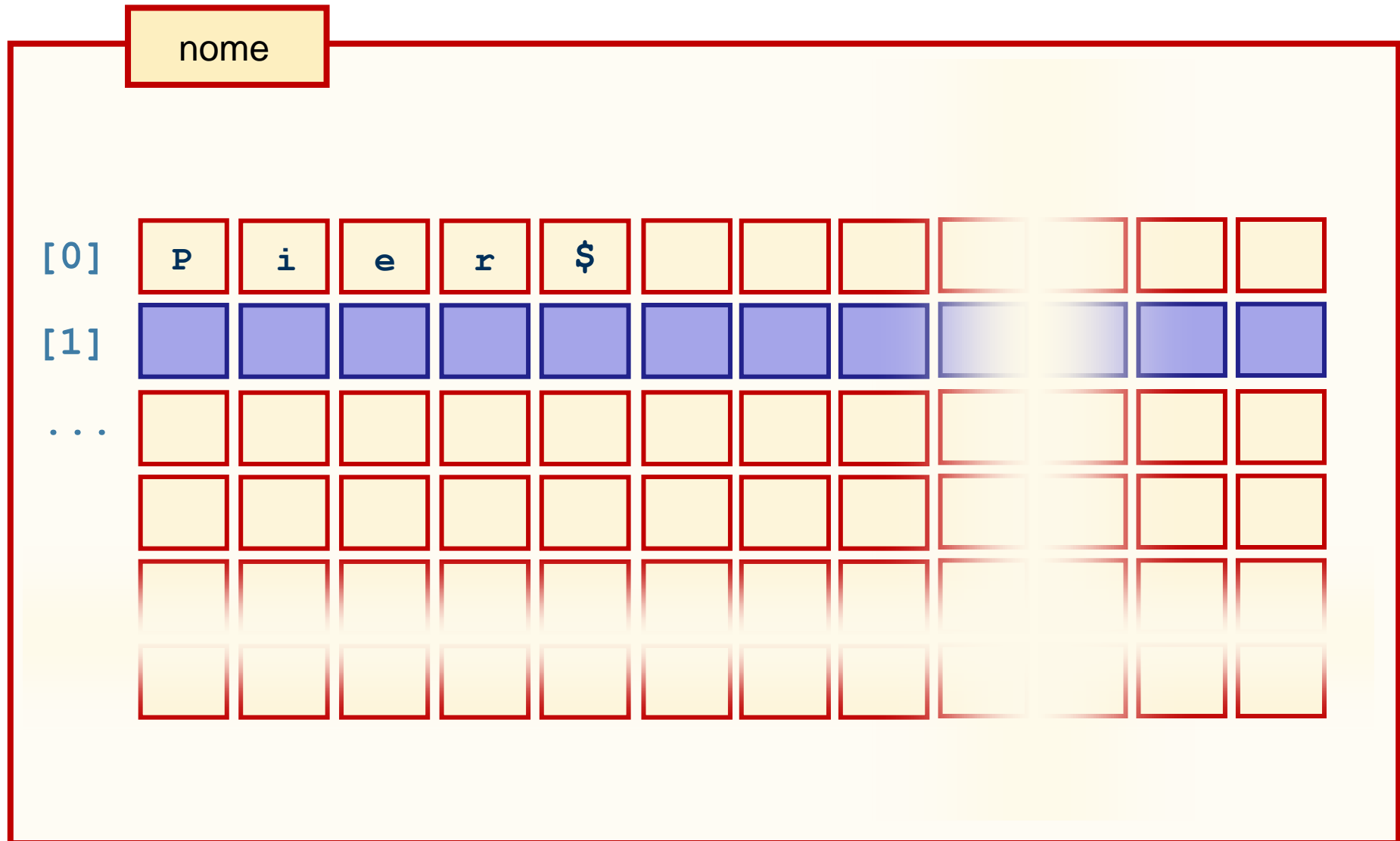
```
...  
pos = 0;  
do  
{ saltaspaзи();  
  i = 0;  
  do  
    { nome[pos][i] = carattere;  
      i++;  
      cin >> carattere;  
    } while (carattere != tappo && carattere != spazio);  
    nome[pos][i] = '$';  
  
  } while (carattere != tappo);  
...
```

nome

P	i	e	r	\$							
L	u	i	g	i	\$						

Della Vigna Pier Luigi

```
...  
pos = 0;  
do  
{ saltaspazi();  
  i = 0;  
  do  
    { nome[pos][i] = carattere;  
      i++;  
      cin >> carattere;  
    } while (carattere != tappo && carattere != spazio);  
  nome[pos][i] = '$';  
  pos ++;  
} while (carattere != tappo);  
...
```



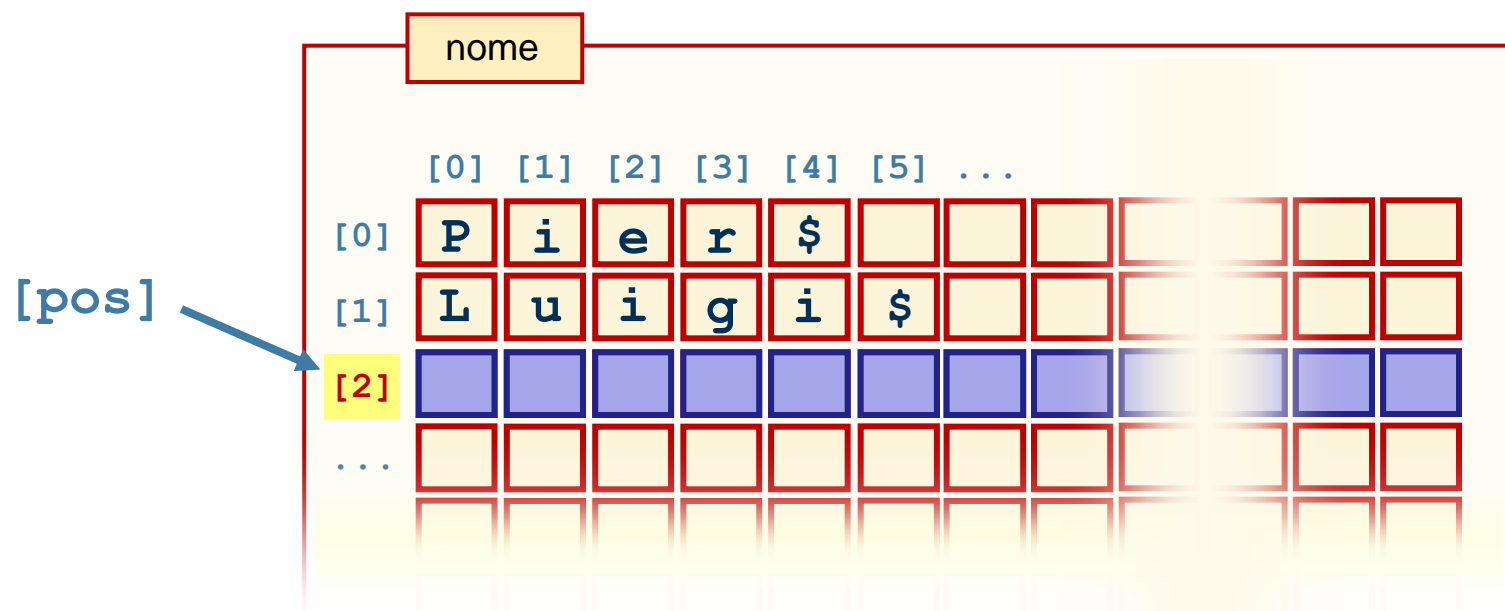

```
<< "Le parti sono separate da uno o più spazi. << endl;

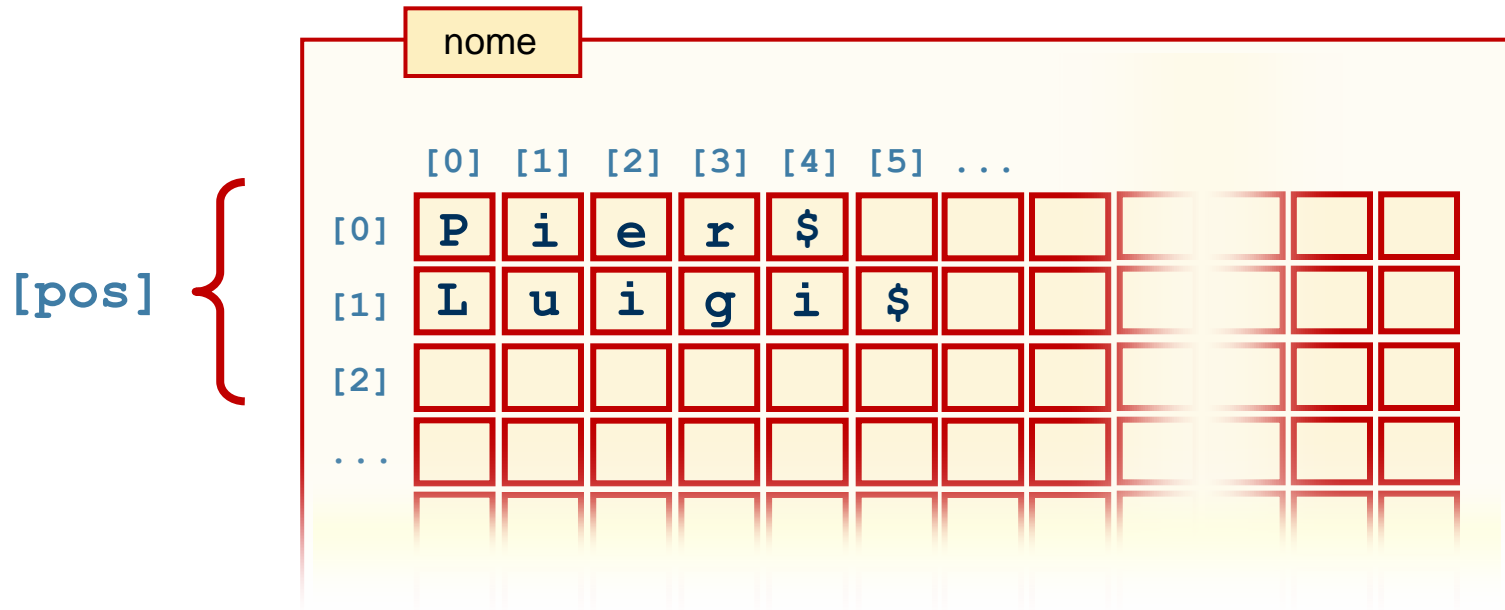
// conserva una parte del nome
pos = 0;
do
{ // salta spazi prima di parte del nome
  saltaspace();
  // leggi e scrivi ciascuna parte del nome
  i = 0;
  do
  { nome[pos][i] = carattere;
    i++;
    cin >> carattere;
  } while (carattere != tappo && carattere != spazio);
  // inserisci il tappo alla fine di ogni parte del nome
  nome[pos][i] = '$';
  // passa alla successiva parte del nome
  pos++;
} while (carattere != tappo);

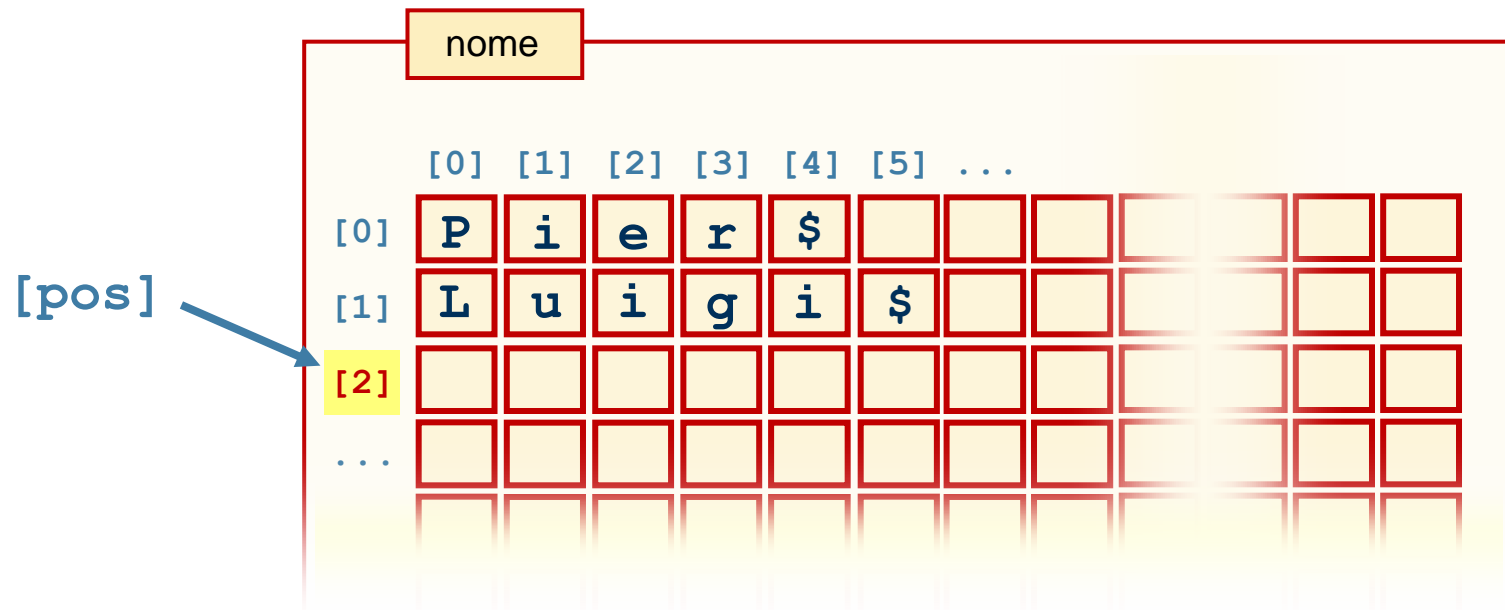
// leggi e scrivi cognome composto da una o più parti
do
{ // salta spazi prima di parte del cognome
  saltaspace();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
```

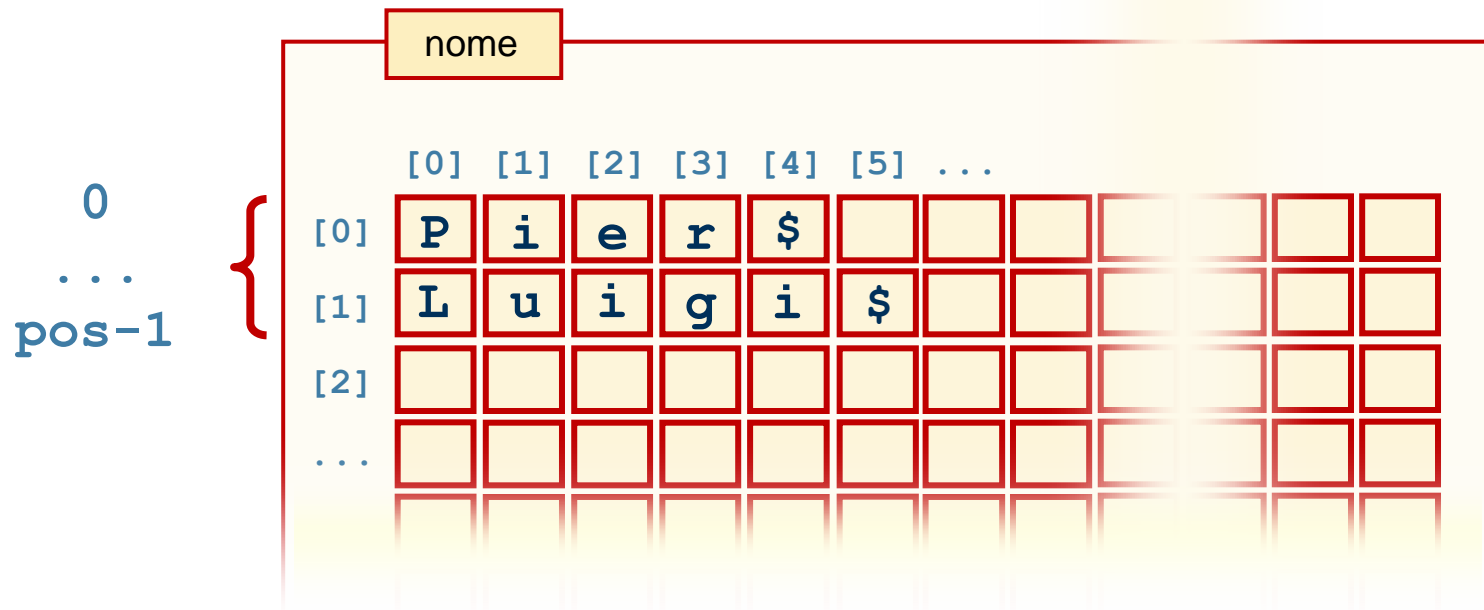
```
do
{ // salta spazi prima di parte del cognome
  saltaspazi();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
    } while (carattere != tappo && carattere != spazio);
  // scrivi spazio
  cout << spazio;
} while (carattere != tappo);

// scrivi i carattere del nome
for (j = 0; j <= i-1; j++)
{ cout << nome[j];
}
}
```

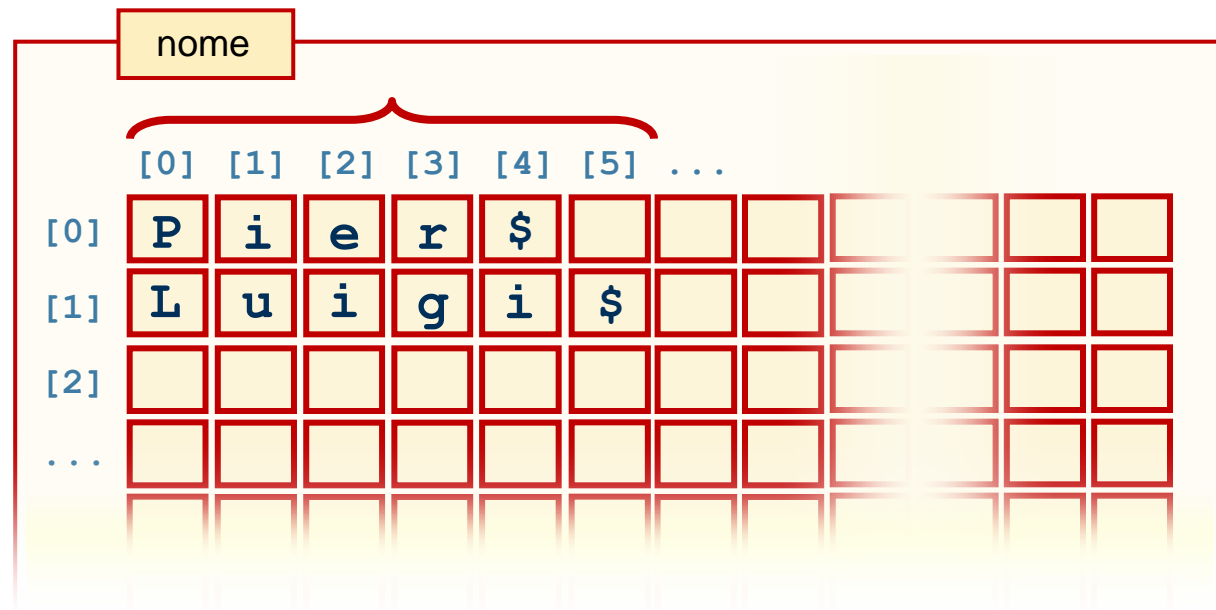




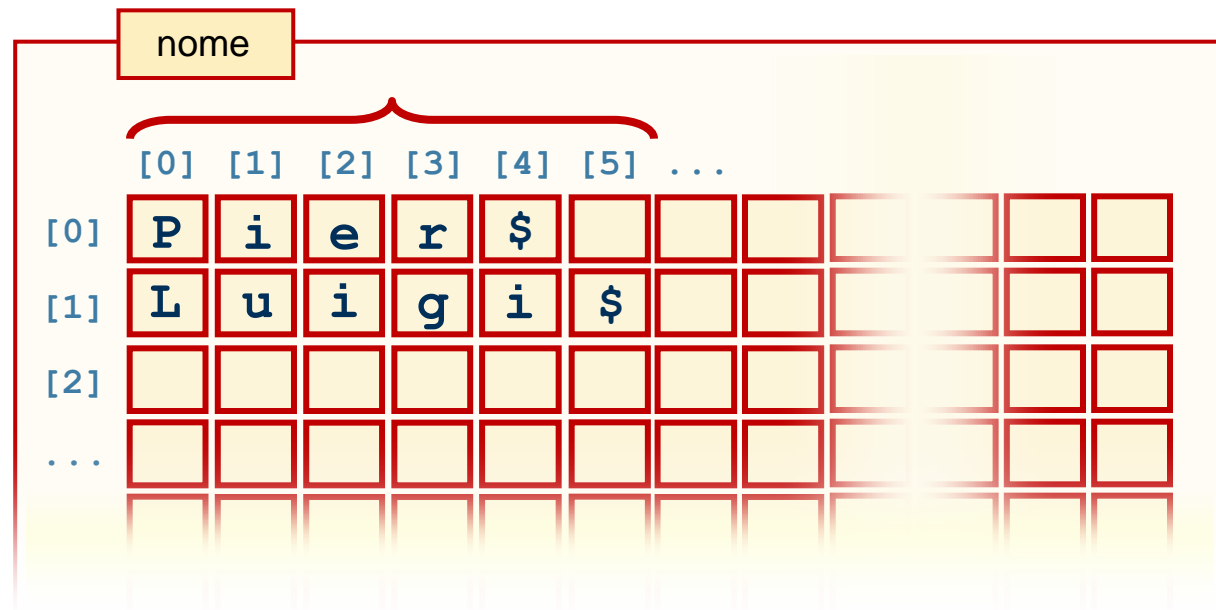




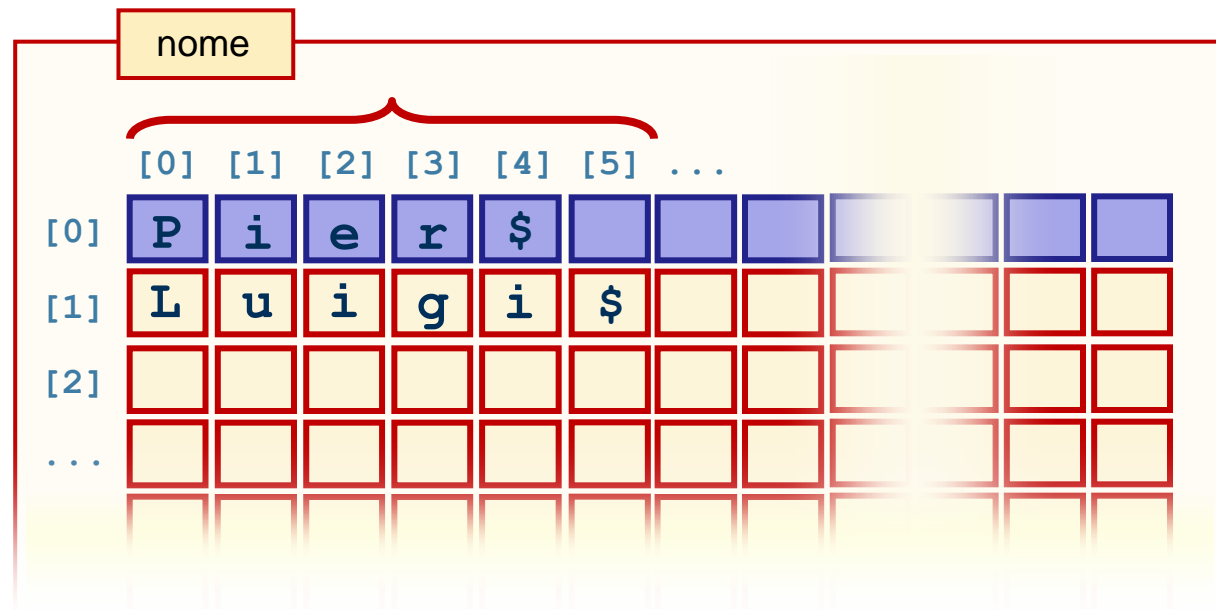
```
// scrivi i caratteri del nome
for (k = 0; k <= pos-1; k++)
{
    ...
}
```



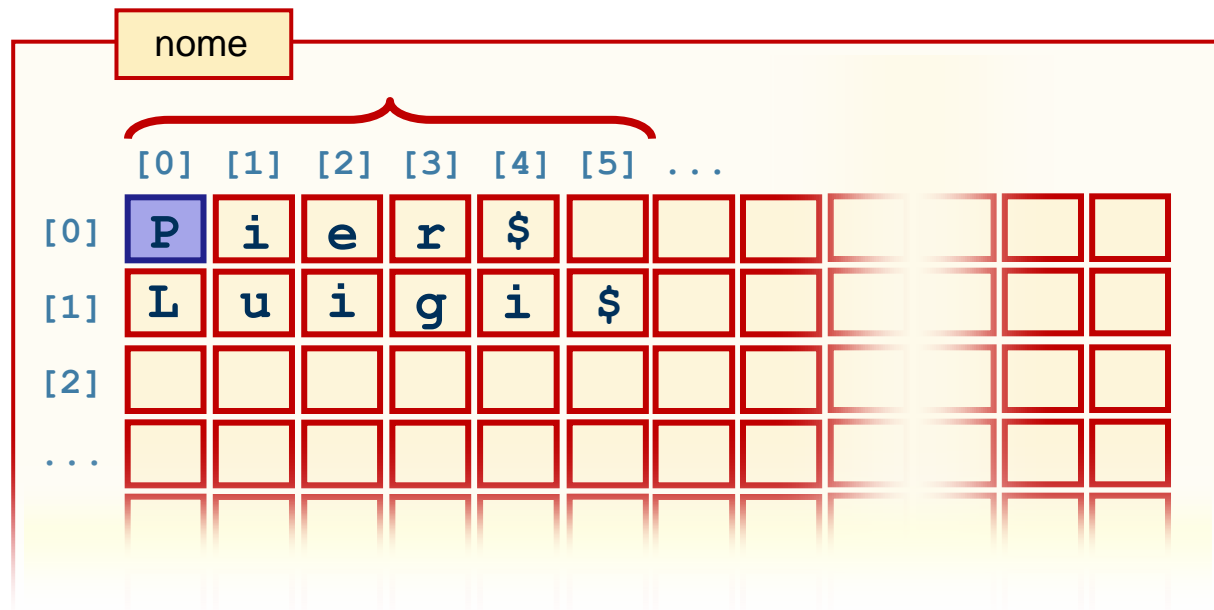
```
// scrivi i caratteri del nome
for (k = 0; k <= pos-1; k++)
{
    ...
}
```



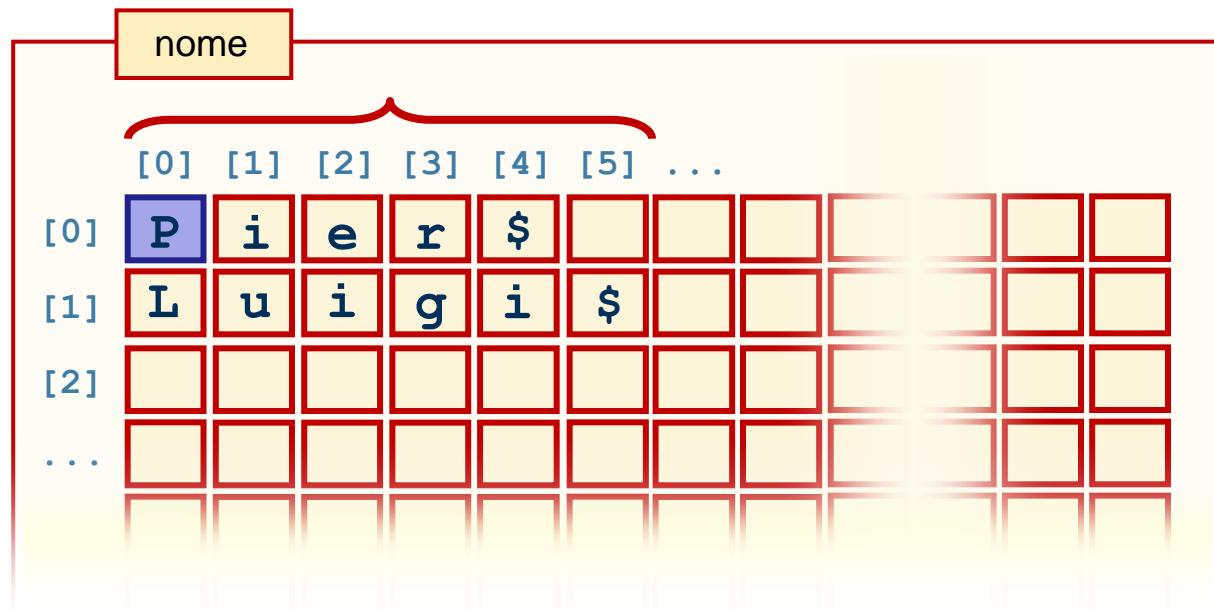
```
// scrivi i caratteri del nome
for (k = 0; k <= pos-1; k++)
{
    ...
    cout << spazio;
}
```

```
// scrivi i caratteri del nome
for (k = 0; k <= pos-1; k++)
{
    ...
    cout << spazio;
}
```



```
// scrivi i caratteri del nome
for (k = 0; k <= pos-1; k++)
{
    j = 0;
    ...
    cout << spazio;
}
```



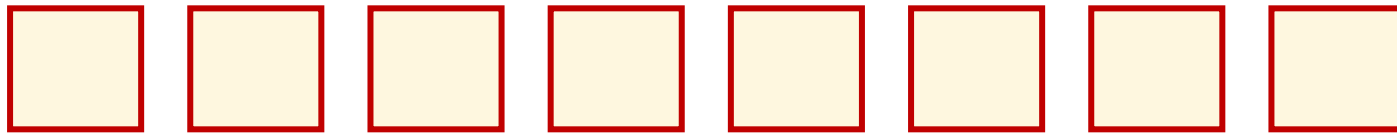
```
// scrivi i caratteri del nome
for (k = 0; k <= pos-1; k++)
{ j = 0;
  while (nome[k][j] != '$')
  { cout << nome[k][j];
    j++;
  }
  cout << spazio;
}
```

```
        // passa alla successiva parte del nome
        pos ++;
    } while (carattere != tappo);

// leggi e scrivi cognome composto da una o piu' parti
do
{ // salta spazi prima di parte del cognome
  saltaspaзи();
  // leggi e scrivi parte del cognome
  do
  { cout << carattere;
    cin >> carattere;
    } while (carattere != tappo && carattere != spazio);
  // scrivi spazio
  cout << spazio
} while (carattere != tappo);

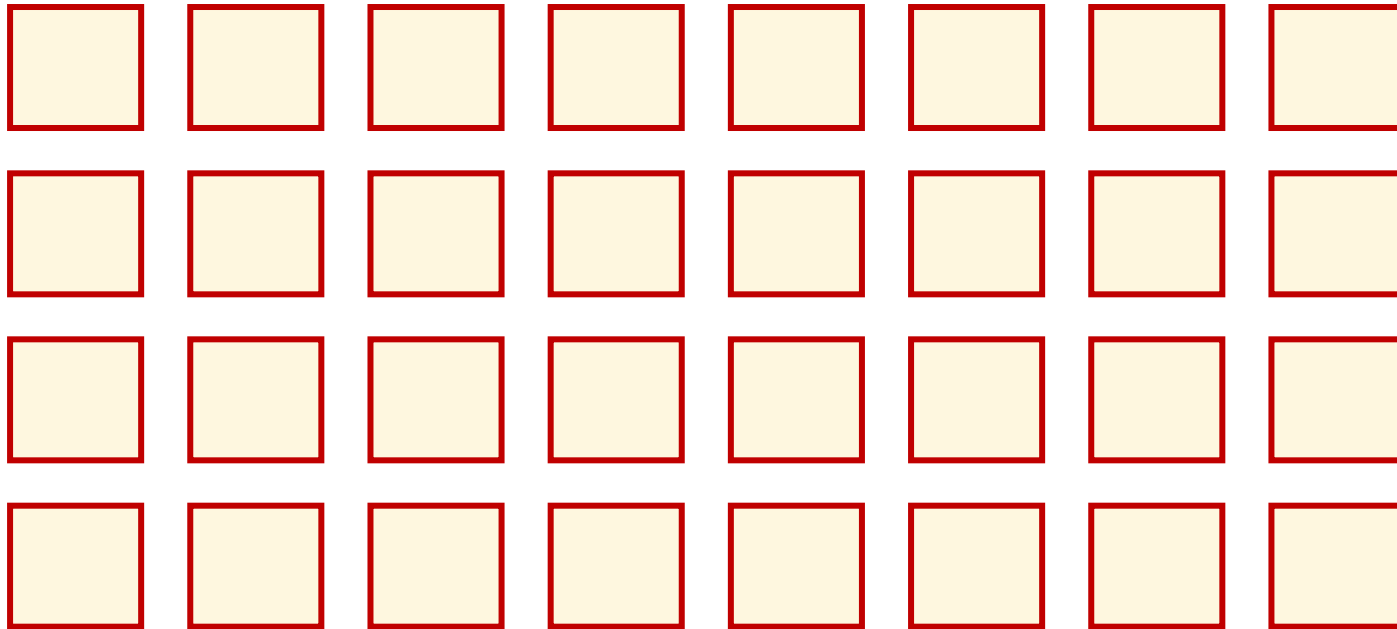
// scrivi i caratteri del nome
for (k = 0; k <= pos-1; k++)
{ j = 0;
  while (nome[k][j] != '$')
  { cout << nome[k][j];
    j++;
  }
  cout << spazio;
}
}
```

Array monodimensionali



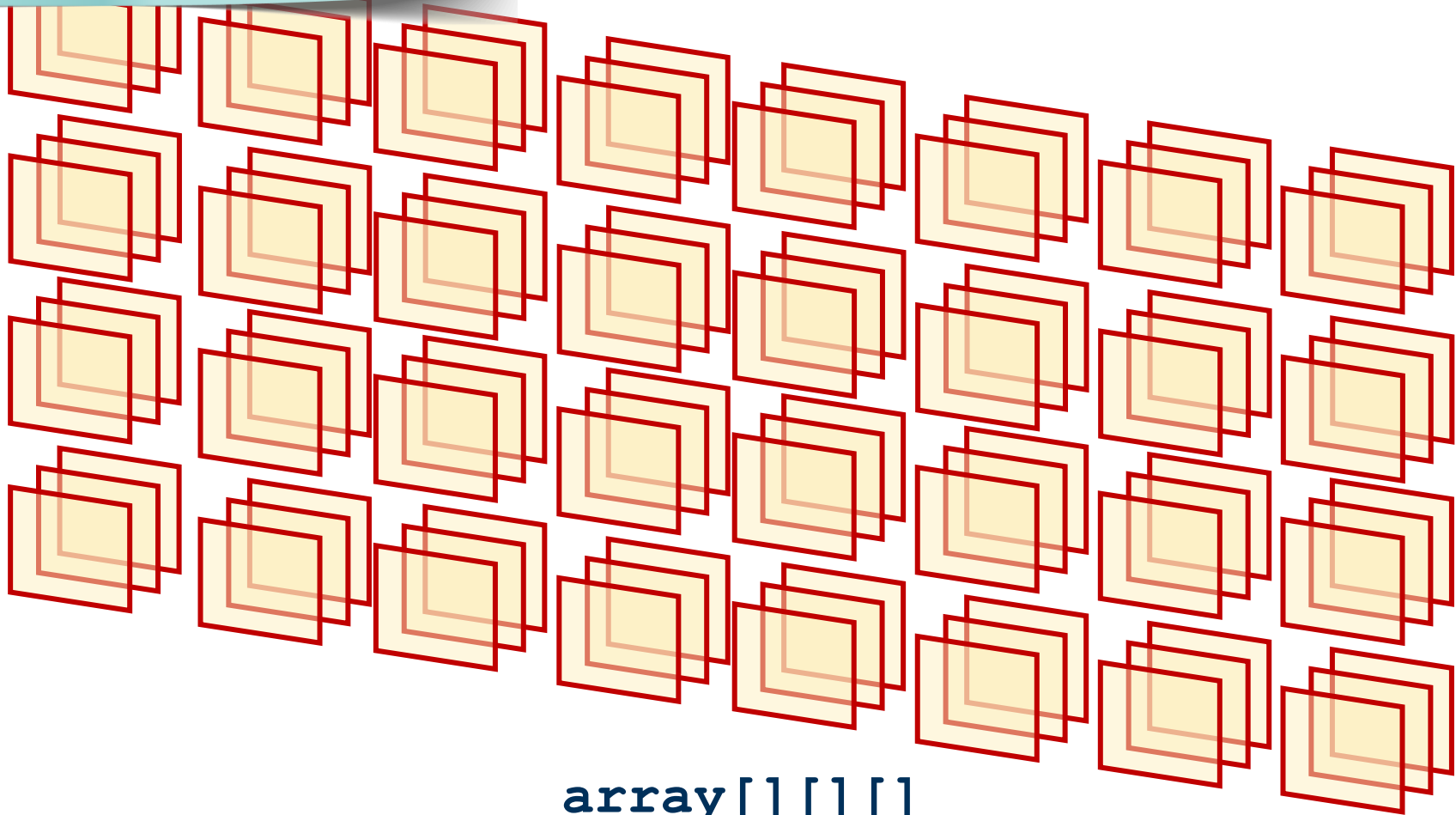
`array[]`

Array bidimensionali

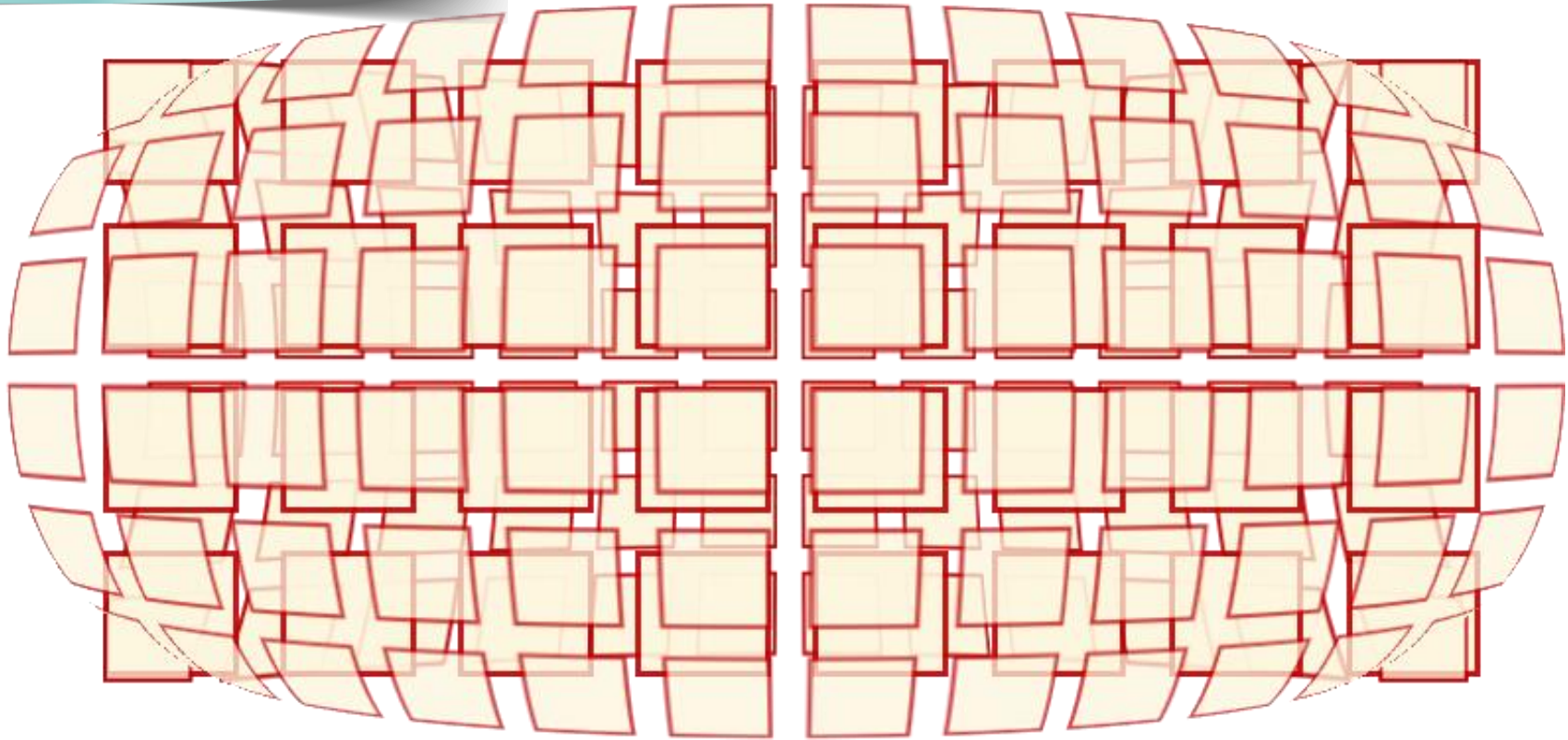


`array[][]`

Array tridimensionali



Array n-dimensionali



`array[][][][][]...`

C++

```
char array[...][...];
```

```
char array[const][const];
```

```
char array[10][20];
```

```
const int lung_max = 20  
char array[10][lung_max];
```