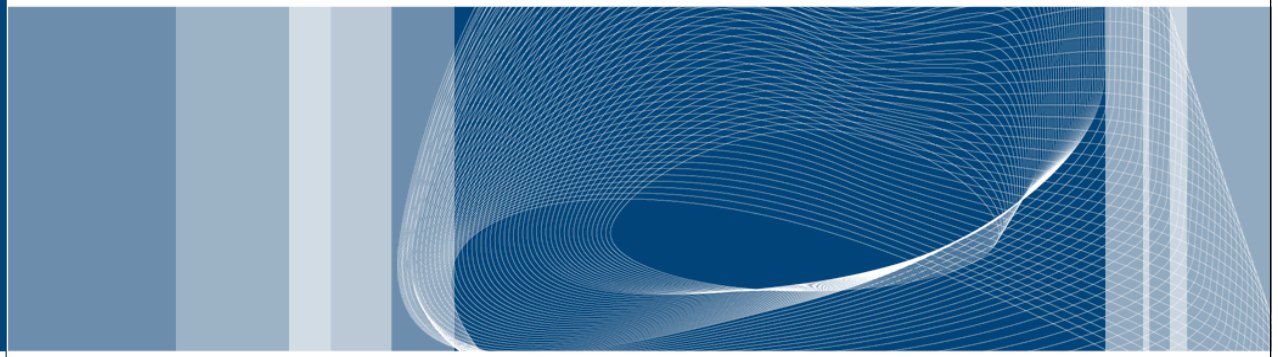
POLITECNICO DI MILANO



Fondamenti di Informatica 2013-2014

Puntatori

Paola Mussida
Area Servizi ICT

Esercizio P_0 - Istruzioni

Sviluppare un programma che codifica in C le seguenti istruzioni:

- ✓ dichiara il tipo di puntatori ad interi typePtrInt
- ✓ dichiara una variabile di tipo typePtrInt e la chiama ptrInt
- ✓ dichiara due variabili intere varInt1 e varInt2
- ✓ dichiara una variabile di tipo puntatore a typePtrInt e la chiama doppioPtrInt

....

Esercizio P_0 - Istruzioni

```
#include <stdio.h>

typedef int *typePtrInt;

void main(){
 typePtrInt ptrInt;

 int varInt1, varInt2;

 int temporanea;

 typePtrInt *doppioPtrInt;
 //int **doppioPtrInt;
```

Esercizio P_0 - Istruzioni

Sviluppare un programma che codifica in C le seguenti istruzioni:

- ✓ assegna a varInt1 un valore digitato da tastiera
- ✓ assegna a ptrInt l'indirizzo di varInt1
- ✓ stampa il contenuto di ciò che viene puntato da ptrInt
- ✓ assegna a varInt1 un nuovo valore digitato da tastiera ma diverso dal precedente (si inserisca nel programma un opportuno controllo)

....

Esercizio P_0 - Istruzioni

```
printf("\n\nInserisci un valore per varInt1\n");

scanf("%d", &varInt1);
ptrInt=&varInt1;

printf("\n\nValore inserito: %d\n",*ptrInt);
printf("\n\nInserisci un valore diverso per varInt1\n");

scanf("%d",&temporanea);
while (temporanea==varInt1)
{
        printf("\n\nIl valore che hai inserito non e'
                                    diverso, riprova.\n");
        scanf("%d",&temporanea);
}
varInt1=temporanea;
```

Esercizio P_0 - Istruzioni

Sviluppare un programma che codifica in C le seguenti istruzioni:

- ✓ stampa il contenuto di cio' che viene puntato da ptrInt
- ✓ assegna a cio' che viene puntato da ptrInt il valore 12
- ✓ stampa il contenuto di varInt1
- ✓ assegna a varInt2 il valore 100
- ✓ assegna a ptrInt l'indirizzo di varInt2
- ✓ stampa il contenuto di cio' che viene puntato da ptrInt e di varInt1
- ✓ assegna a doppioPtrInt l'indirizzo di ptrInt

....

```
//uguale all'istruzione di prima
printf("\n\nValore di ''*PtrInt'': %d\n",*ptrInt);

*ptrInt=12; //varInt1=12;

printf("\n\nValore di ''varInt1'': %d\n",varInt1);

varInt2=100;

ptrInt=&varInt2; // e non *ptrInt=&varInt2;

printf("\n\nValore di ''*ptrInt'': %d\n",*ptrInt);
printf("\n\nValore di ''varInt1'': %d\n",varInt1);

doppioPtrInt=&ptrInt;
```

Sviluppare un programma che codifica in C le seguenti istruzioni:

- ✓ effettua la seguente operazione di stampa: printf("%d", **doppioPtrInt);
- ✓ effettua la seguente operazione di assegnamento: **doppioPtrInt = 50;

```
    printf("%d",**doppioPtrInt);
    **doppioPtrInt=50;

}
```

Un semplice programma di esempio sui puntatori.

Esercizio P_1 - Esempio

```
int number = 0;


/* Un puntatore che puo'
puntare ad un int: */

int *pointer = NULL;
```

Esercizio P_1 - Esempio

```
int main()
{
   number = 10;
// Stampa l'indirizzo di number:
   printf("\nnumber's address: %p", &number);

// Stampa il valore:
   printf("\nnumber's value: %d", number);

//Memorizza l'indirizzo di "number" in "pointer"
   pointer = &number;

//Stampa l'indirizzo
   printf("\npointer's address: %p", &pointer);
```

Esercizio P_1 - Esempio


```
//Stampa il valore (in questo caso e' un indirizzo)
   printf("\npointer's value: %p", pointer);

//Valore all'indirizzo
   printf("\nvalue pointed to: %d", *pointer);

//Memorizza 15 in 'number'
   *pointer = 15;

//Stampa il valore puntato
   printf("\nvalue pointed to: %d", *pointer);

//Stampa il valore
   printf("\nnumber's value: %d", number);
}
```

Un programma di esempio sull'aritmetica dei puntatori.

Esercizio P_2 - Esempio aritmetico

```
#include <stdio.h>


char strg[40], *there, one, two;

int *pt, list[100], index;
```

Esercizio P_2 - Esempio aritmetico

```
int main()
{
strcpy(strg,"This is a character string.");
one = strg[0];
two = *strg; //"one" e "two" sono uguali

printf("The first output is %c %c\n",one,two);
                        //The first output is T T
one = strg[8];

two = *(strg+8); // "one" e "two" sono uguali
                        //The second output is a a
printf("The second output is %c %c\n",one,two);
```

Esercizio P_2 - Esempio aritmetico

```
  there = strg+10;
                //"strg+10" e' uguale a "&strg[10]"
  printf("The third output is %c\n",strg[10]);
                          //The third output is c
  printf("The fourth output is %c\n",*there);
                          //The fourth output is c
  for (index = 0; index<100; index++)
      list[index] = index + 100;

  pt = list + 27;

  printf("The fifth output is %d\n",list[27]);
                          //The fifth output is 127
  printf("The sixth output is %d\n",*pt);
}                         //The sixth output is 127
```