

Esempio

Scrivere una procedura che sostituisce alla variabile x il suo valore assoluto: void abs(int *x)

```
void abs(int *x) {  
    if (*x < 0)  
        *x = -(*x);  
}
```

- Nel chiamante si utilizzano chiamate del tipo abs(&a) per ottenere l'effetto di rimpiazzare il valore della variabile a con il suo valore assoluto.

Esempio: parametri di tipo vettore

- Quando si passa un vettore come parametro ad una funzione, in realtà si sta passando l'indirizzo dell'elemento di indice 0: il parametro formale deve essere di tipo **puntatore** al tipo degli elementi del vettore.
- Scrivere la procedura che stampa tutti gli elementi di un vettore, i cui elementi sono stati letti da input.

```
main() {  
    int vet[5];  
    leggiVettore(vet,5);  
    stampaVettore(vet, 5);  
    ...  
}
```

```
void stampaVettore(int *v, int dim) {  
    int i;  
    for (i = 0; i < dim; i++)  
        printf("v[%d]: %d\n", i, v[i]);  
}
```

```
void leggiVettore(int v[], int dim) {  
    int i;  
    for (i = 0; i < dim; i++) {  
        printf("Immettere l'elemento di indice %d: ", i);  
        scanf("%d", &v[i]);  
    }  
}
```

Esercizio d'esame n. 1

- Scrivere l'algoritmo ed il codice C di un programma che legga dallo standard input N interi, li memorizzi in un array e li ordini attraverso una funzione di ordinamento predefinita.
- Scrivere gli interi ordinati su un file di testo, il cui nome è letto da linea di comando.

c:> OrdinalInteri fileoutput.txt

Soluzione esercizio n. 1

```
#include <stdio.h>
#define N 20

void quickSort(int v[ ], int inizio, int fine);

main(int argc, char * argv[]) {
    int v[N], i;
    char linea[L];
    FILE *fp;

    if ( (FIN=fopen( argv[1] ,"w")) == NULL ) {
        printf("Non posso aprire il file di origine");
        return;
    }
}
```

```
for(i=0; i<N; i++) {
    printf("inserire intero ");
    scanf("%d",&v[i]);
}

ordinaVettore(v,N);

for(i=0; i<N; i++) {
    fprintf(fp,"/t %d \n",v[i]);
}

fclose(fp);
}
```

Esercizio d'esame n. 2

- Scrivere l'algoritmo ed il codice C di un programma che conti il numero di righe di un file di testo (il cui nome è fornito a linea di comando) e aggiunga al file *numerorighe.txt* il nome del file letto e il numero di righe di cui è composto.

```
c:> contaRighe fileinput.txt
```

Soluzione esercizio n. 2

```
#include <stdio.h>
#define L 200

main(int argc, char * argv[]) {
    int numerorighe=0;
    char linea[L];
    FILE *FIN, *FOUT;

    if ( (FIN=fopen(argv[1],"r")) == NULL )
        printf("Non posso aprire il file di
            origine");
    else
```

```
if ( (FOUT=fopen("numerorighe.txt ","a")) == NULL )
    {
        printf("Non posso aprire il file destinazione");
        fclose(FIN);
    }
    else {
        while (fgets(linea, L, FIN) != NULL)
            numerorighe++;
        fprintf (FOUT, "%s /t %d", argv[1],numerorighe);
        fclose(FIN);
        fclose(FOUT);
    }
}
```