

## E11 – Esercizi su Strutture, Unioni ed Enumerazioni in C

**Esercizio 1.** Definire una struttura di nome **data** con i campi **giorno mese** ed **anno** tutti di tipo **int**. Scrivere poi una funzione che riceve un parametro di tipo **data** e stampa la data passata come parametro nella forma estesa in cui il giorno è rappresentato con un numero da 1 a 31, il nome del mese è riportato per esteso e l'anno è riportato come numero di quattro cifre. Qualora la data passata come parametro contenga un anno di sole due cifre, esso va considerato un anno del XX secolo se superiore a 16 e un anno del XXI secolo se minore o uguale a 16.

**Esercizio 2.** Definire una struttura di nome **complesso** dotata dei campi **re** e **im** per rappresentare numeri complessi. Scrivere una funzione per la risoluzione di equazioni di secondo grado (v. anche l'Esercizio 5 di E7 e gli esercizi 1 e 2 di E9) della forma  $ax^2 + bx + c = 0$ . La funzione riceve tre parametri (di tipo **double**) passati per valore che rappresentano i tre coefficienti  $a$ ,  $b$  e  $c$  e due parametri (di tipo **complesso**) passati per riferimento in cui scrivere le soluzioni dell'equazione.

Scrivere poi un programma che fa inserire all'utente i coefficienti di un'equazione di secondo grado e stampa le sue soluzioni.

**Esercizio 3.** Definire una struttura di nome **punto** con i campi **x** e **y** di tipo **double**. Scrivere una funzione **distanza** che dati due parametri di tipo **punto** restituisce la distanza Euclidea tra i due punti. Scrivere poi una funzione **lunghezzaComplessiva** che riceve come parametro un array di punti che rappresentano una poligonale e calcola la lunghezza complessiva della poligonale.

Scrivere un programma che fa inserire all'utente una sequenza di punti (la lunghezza della sequenza è scelta dall'utente) e stampa la lunghezza complessiva della poligonale rappresentata dai punti inseriti.

**Esercizio 4.** Assumendo di aver definito la struttura di nome **punto**, definire una struttura **cerchio** i cui campi sono **centro** (di tipo **punto**) e **raggio** (di tipo **double**). Utilizzando la funzione **distanza** dell'esercizio precedente scrivere una funzione che riceve un parametro **c** di tipo **cerchio** e un parametro **p** di tipo **punto** e restituisce 1 se il punto **p** è all'interno del cerchio **c**, 0 altrimenti. Scrivere poi un programma che utilizza la funzione scritta.

**Esercizio 5.** Assumendo di aver definito la struttura di nome **punto**, definire una struttura **retta** i cui campi sono **m** (di tipo **double**) e **q** (di tipo **double**), rappresentanti il coefficiente angolare e il termine noto dell'equazione della retta  $y = mx + q$ . Scrivere una funzione che riceve come parametri una retta **r** ed un punto **p** e restituisce la retta perpendicolare a **r** passante per **p**. Scrivere poi un programma che utilizza la funzione scritta.

*Nota: Qualora la retta sia verticale (cioè avente equazione  $y=c$ ) il valore di  $m$  deve essere impostato a **INFINITY** (tale costante è definita nella libreria `math.h`) e  $q$  deve essere uguale a  $c$ .*

**Esercizio 6.** Assumendo di aver definito la struttura di nome **punto**, definire una struttura **parabola** i cui campi sono **a**, **b** e **c** (tutti di tipo **double**) rappresentanti i coefficienti dell'equazione della parabola  $y = ax^2 + bx + c$ . Scrivere una funzione che riceve come parametro una parabola **p** e restituisce un **punto** che è il fuoco della parabola. Scrivere poi un programma che utilizza la funzione scritta.

## Soluzioni

### Esercizio 1 - svolgimento.

```
#include <stdio.h>

typedef struct {
    int giorno;
    int mese;
    int anno;
} data;

void stampaEstesa(data d);

int main(){
    data d;
    printf("Inserisci una data (giorno, mese ed anno in cifre)\n");
    scanf("%d%d%d", &d.giorno, &d.mese, &d.anno);
    printf("La data inserita e':\n");
    stampaEstesa(d);
}

void stampaEstesa(data d){
    printf("%d ", d.giorno);

    char * mesi[12]={"gennaio", "febbraio", "marzo", "aprile",
    "maggio", "giugno", "luglio", "agosto", "settembre", "ottobre",
    "novembre", "dicembre"};
    printf("%s ", mesi[d.mese-1]);

    int anno=d.anno;
    if(d.anno < 1000){
        if(d.anno<=16)
            anno=2000+anno;
        else
            anno=1900+anno;
    }

    printf("%d ", anno);
}
```

### Esercizio 2 - svolgimento.

```
#include <stdio.h>
#include <math.h>

typedef struct {
    double re;
    double im;
} complesso;

void soluzioni(double a, double b, double c, complesso * sol1, complesso
* sol2);

int main(void){

    printf("Inserisci i tre coefficienti dell'equazione\n");
    double a, b, c;
    scanf("%lf%lf%lf",&a, &b, &c);
```

```

    complesso sol1, sol2;
    soluzioni(a, b, c, &sol1, &sol2);
    printf("Le soluzioni dell'equazione sono: %g + i %g e %g + i %g\n",
        sol1.re, sol1.im, sol2.re, sol2.im);
}

void soluzioni(double a, double b, double c, complesso * sol1, complesso
* sol2){
    double delta=b*b-4*a*c;
    if(delta>0){
        (*sol1).re=(-b+sqrt(delta))/(2*a);
        (*sol2).re=(-b-sqrt(delta))/(2*a);
        (*sol1).im=0;
        (*sol2).im=0;
    }else{
        (*sol1).re=-b/(2*a);
        (*sol1).im=sqrt(-delta)/(2*a);
        (*sol2).re=(*sol1).re;
        (*sol2).im=-sqrt(-delta)/(2*a);
    }
}

```

### Esercizio 3 - svolgimento.

```

#include <stdio.h>
#include <math.h>

typedef struct {
    double x;
    double y;
} punto;

double distanza(punto p1, punto p2);
double lunghezzaComplessiva(punto poligonale[], int dim);

int main(){
    int n;
    printf("Quanti punti vuoi inserire?\n");
    scanf("%d", &n);
    punto poligonale[n];

    for(int i=0;i<n;i++){
        printf("Inserisci il punto n. %d\n", i);
        scanf("%lf%lf", &poligonale[i].x, &poligonale[i].y);
    }

    double lunghezza=lunghezzaComplessiva(poligonale, n);
    printf("La lunghezza della poligonale e': %g\n", lunghezza);
}

double distanza(punto p1, punto p2){
    return sqrt(pow(p1.x-p2.x,2)+pow(p1.y-p2.y,2));
}

double lunghezzaComplessiva(punto poligonale[], int dim){
    double lung=0.0;
    for(int i=0;i<dim-1;i++)

```

```

        lung+=distanza(poligonale[i], poligonale[i+1]);
    return lung;

}

```

#### Esercizio 4 - svolgimento.

```

#include <stdio.h>
#include <math.h>

typedef struct {
    double x;
    double y;
} punto;

typedef struct {
    punto centro;
    double raggio;
} cerchio;

double distanza(punto p1, punto p2);

int contiene(cerchio c, punto p);

int main(){
    cerchio c;
    printf("Inserisci le coordinate del centro\n");
    scanf("%lf%lf", &c.centro.x, &c.centro.y);

    printf("Inserisci il raggio\n");
    scanf("%lf", &c.raggio);

    punto p;
    printf("Inserisci le coordinate del punto\n");
    scanf("%lf%lf", &p.x, &p.y);

    if(contiene(c,p))
        printf("Il punto e' contenuto nel cerchio\n");
    else
        printf("Il punto NON e' contenuto nel cerchio\n");
}

double distanza(punto p1, punto p2){
    return sqrt(pow(p1.x-p2.x,2)+pow(p1.y-p2.y,2));
}

int contiene(cerchio c, punto p){
    int contenuto=0;
    if(distanza(c.centro,p)<c.raggio)
        contenuto=1;
    return contenuto;
}

```

#### Esercizio 5 - svolgimento.

```

#include <stdio.h>
#include <math.h>

typedef struct {

```

```

    double x;
    double y;
} punto;

typedef struct {
    double m;
    double q;
} retta;

retta perpendicolare(retta r, punto p);

int main(){
    retta r;
    printf("Inserisci coeff. angolare e termine noto della retta\n");
    scanf("%lf%lf", &r.m, &r.q);

    punto p;
    printf("Inserisci le coordinate di un punto\n");
    scanf("%lf%lf", &p.x, &p.y);

    retta s=perpendicolare(r,p);
    if(s.m==INFINITY)
        printf("Retta perpendicolare x=%g\n", s.q);
    else if(s.m==0)
        printf("Retta perpendicolare y=%g\n", s.q);
    else
        printf("Retta perpendicolare y=%g x + %g\n",s.m, s.q);
}

retta perpendicolare(retta r, punto p){
    retta s;
    if(r.m==0){
        s.m=INFINITY;
        s.q=p.x;
    }else if(r.m==INFINITY){
        s.m=0;
        s.q=p.y;
    }else{
        s.m=-1/r.m;
        s.q=1/r.m*p.x+p.y;
    }

    return s;
}

```

### Esercizio 6 - svolgimento.

```

#include <stdio.h>
#include <math.h>

typedef struct {
    double x;
    double y;
} punto;

typedef struct {

```

```
double a;
double b;
double c;
} parabola;

punto fuoco(parabola p);

int main(){
    parabola p;
    printf("Inserisci i coefficienti dell'equazione della parabola\n");
    scanf("%lf%lf%lf", &p.a, &p.b, &p.c);

    punto f=fuoco(p);
    printf("Il fuoco della variabile e' (%g,%g)\n", f.x, f.y);
}

punto fuoco(parabola p){
    punto f;
    f.x=-p.b/(2*p.a);
    f.y=(1-p.b*p.b+4*p.a*p.c)/(4*p.a);
    return f;
}
```