

PROVA PRATICA DI FONDAMENTI DI INFORMATICA L-A
Prof. MICHELA MILANO – 25 NOVEMBRE 2004
COMPITO A

Si scriva un programma C che permetta di inserire i dati relativi a correntisti di una banca (si supponga per comodità che saranno inseriti al massimo 4 correntisti). Ogni correntista è caratterizzato dai seguenti dati:

- **Cognome:** una stringa di lunghezza massima di 20 caratteri, non contenente spazi.
- **NumeroConto:** intero
- **Importo:** float
- **PercentualeInteresse:** intero positivo inferiore a 5. Si controlli che l'interesse sia compreso tra 0 e 5. NOTA: se questo campo contiene il numero intero 2, significa che l'interesse è del 2%.

Tali dati vanno inseriti in un array di strutture `struct cliente`. Ogni struttura contiene questi quattro campi.

Letto da tastiera un numero di conto corrente, il programma deve stampare a video il nuovo importo a fine anno del conto corrente corrispondente al numero inserito, calcolato come il vecchio importo più l'interesse maturato.

A tal fine si usi una funzione che prenda il numero di conto corrente letto da utente, l'array di strutture e fornisca in uscita il nuovo importo

```
float nuovoimporto(int numeroconto, struct cliente V[], int Dim);
```

Tale funzione scorrendo l'array V di strutture cliente di dimensione Dim, identifica la cella in cui il campo numeroconto coincide con quello passato alla funzione e calcola il nuovo importo.

FACOLTATIVO:

Si organizzi il progetto su più file in modo da separare il main dalle funzioni e si crei il corrispondente file header.

NOTA: Si consegnino in questo caso TUTTI i FILE .c e il .h

```

#include <stdio.h>

struct cliente {char Cognome[21];
                int NumeroConto;
                float Importo;
                int PercentualeInteresse;
                };

float nuovoimporto(int numeroconto, struct cliente V[], int Dim)
{int i;
  for (i=0; i<Dim; i++)
    if (V[i].NumeroConto==numeroconto)
      return (V[i].Importo + V[i].Importo*(V[i].PercentualeInteresse/100.0));
  return 0;
}

void main()
{int i,n;
  struct cliente V[4];
  do
    {printf("Numero di clienti (max. 4) :");
     scanf("%d", &n);
    }
  while ((n<0) || (n>4));

  for (i=0; i<n; i++)
    {printf("Cognome del cliente #%d : ",i);
     scanf("%s", V[i].Cognome);
     printf("Numero conto del cliente #%d : ",i);
     scanf("%d", &V[i].NumeroConto);
     printf("Importo del cliente #%d : ",i);
     scanf("%f", &V[i].Importo);
     do
       {printf("Percentuale interesse del cliente #%d (0..5) : ",i);
        scanf("%d", &V[i].PercentualeInteresse);
       }
     while ((V[i].PercentualeInteresse>5) || (V[i].PercentualeInteresse<0));
    }

  int conto;
  printf("Numero conto da aggiornare : ");
  scanf("%d", &conto);

  float nuovo=nuovoimporto(conto, V, n);

  if (nuovo) printf("Il nuovo importo del conto %d e' %f.", conto, nuovo);
  else printf("conto non esistente");
}

```

PROVA PRATICA DI FONDAMENTI DI INFORMATICA L-A
Prof. MICHELA MILANO – 25 NOVEMBRE 2004
COMPITO B

Si scriva un programma C che permetta di inserire i dati relativi a correntisti di una banca (si supponga per comodità che saranno inseriti al massimo 5 correntisti). Ogni correntista è caratterizzata dai seguenti dati:

- **Cognome:** una stringa di lunghezza massima di 20 caratteri, non contenente spazi.
- **NumeroConto:** intero
- **Importo:** float

Tali dati vanno inseriti in un array di strutture `struct conto` contenente questi campi.

Si richieda all'utente di inserire una percentuale di interesse come intero positivo minore di 6 (si controlli che tale percentuale sia compresa tra 0 e 6). NOTA: se l'intero positivo inserito è ad esempio 3, ciò significa che la percentuale di interesse è del 3%.

Il programma deve modificare l'importo di tutti i conti correnti nell'array calcolando il nuovo importo come il vecchio importo più l'interesse maturato.

A tal fine si usi una procedura:

```
void aggiornaimporti(int percentuale, struct conto V[], int Dim);
```

Tale procedura, per ogni cella dell'array V di dimensione Dim di strutture conto, calcola il nuovo importo calcolato come vecchio importo più l'interesse e aggiorna l'array di conseguenza.

FACOLTATIVO:

Si organizzi il progetto su più file in modo da separare il main dalle funzioni e si crei il corrispondente file header.

NOTA: Si consegnino in questo caso TUTTI i FILE .c e il .h

```

#include <stdio.h>

struct conto {char Cognome[21];
              int NumeroConto;
              float Importo;
              };

void aggiornaimporti(int percentuale, struct conto V[], int Dim)
{int i;
  for (i=0; i<Dim; i++)
    V[i].Importo=V[i].Importo + V[i].Importo*(percentuale/100.0);
}

void main()
{int i,n;
  struct conto V[5];
  do
    {printf("Numero di clienti (max. 5) :");
     scanf("%d", &n);
    }
  while ((n<0) || (n>5));

  for (i=0; i<n; i++)
    {printf("Cognome del cliente #%d : ",i);
     scanf("%s", V[i].Cognome);
     printf("Numero conto del cliente #%d : ",i);
     scanf("%d", &V[i].NumeroConto);
     printf("Importo del cliente #%d : ",i);
     scanf("%f", &V[i].Importo);
    }

  int interesse;
  do
    {printf("Inserire l'interesse (0..6) : ");
     scanf("%d", &interesse);
    }
  while ((interesse>6) || (interesse<0));

  aggiornaimporti(interesse, V, n);

  for (i=0; i<n; i++)
    printf("Utente %s\tConto %d\tImporto %f\n", V[i].Cognome, V[i].NumeroConto,
          V[i].Importo);
}

```

PROVA PRATICA DI FONDAMENTI DI INFORMATICA L-A
Prof. MICHELA MILANO – 25 NOVEMBRE 2004
COMPITO C

Si scriva un programma C che permetta di inserire i dati relativi a correntisti di una banca (si supponga per comodità che saranno inseriti al massimo 4 correntisti). Ogni persona è caratterizzata dai seguenti dati:

- **Cognome:** una stringa di lunghezza massima di 20 caratteri, non contenente spazi.
- **NumeroConto:** intero
- **Importo:** float
- **PercentualeInteresse:** intero positivo inferiore a 10. Si controlli che l'interesse sia compreso tra 0 e 10. NOTA: se questo campo contiene il numero intero 4, significa che l'interesse è del 4%.

Tali dati vanno inseriti in un array di strutture `struct conto` contenente questi campi.

Dopo la lettura dei dati, il programma deve aggiornare ogni cella dell'array calcolando il nuovo importo da inserire come somma tra l'importo presente e l'interesse maturato
A tal fine si usi una procedura

```
void calcolointeressi(struct conto V[], int Dim);
```

Tale procedura, per ogni cella dell'array V di strutture cliente, calcola il nuovo importo calcolato come vecchio importo più l'interesse maturato e aggiorna l'array.

FACOLTATIVO:

Si organizzi il progetto su più file in modo da separare il main dalle funzioni e si crei il corrispondente file header.

NOTA: Si consegnino in questo caso TUTTI i FILE .c e il .h

```

#include <stdio.h>

struct conto {char Cognome[21];
              int NumeroConto;
              float Importo;
              int PercentualeInteresse;
              };

void calcolointeressi(struct conto V[], int Dim)
{int i;
  for (i=0; i<Dim; i++)
    V[i].Importo+=V[i].Importo*(V[i].PercentualeInteresse/100.0);
}

void main()
{int i,n;
  struct conto V[4];
  do
    {printf("Numero di clienti (max. 4) :");
     scanf("%d", &n);
    }
  while ((n<0) || (n>4));

  for (i=0; i<n; i++)
    {printf("Cognome del cliente #%d : ",i);
     scanf("%s", V[i].Cognome);
     printf("Numero conto del cliente #%d : ",i);
     scanf("%d", &V[i].NumeroConto);
     printf("Importo del cliente #%d : ",i);
     scanf("%f", &V[i].Importo);
     do
       {printf("Percentuale interesse del cliente #%d (0..10) : ",i);
        scanf("%d", &V[i].PercentualeInteresse);
       }
     while ((V[i].PercentualeInteresse>10) || (V[i].PercentualeInteresse<0));
    }

  calcolointeressi(V, n);

  for (i=0; i<n; i++)
    printf("Utente %s\tConto %d\tImporto %f\n", V[i].Cognome, V[i].NumeroConto,
          V[i].Importo);
}

```

PROVA PRATICA DI FONDAMENTI DI INFORMATICA L-A
Prof. MICHELA MILANO – 25 NOVEMBRE 2004
COMPITO D

Si scriva un programma C che permetta di inserire i dati relativi a correntisti di una banca (si supponga per comodità che saranno inseriti esattamente 3 correntisti). Ogni correntista è caratterizzata dai seguenti dati:

- **Cognome:** una stringa di lunghezza massima di 20 caratteri, non contenente spazi.
- **NumeroConto:** intero
- **Importo:** float
- **PercentualeInteresse:** intero positivo inferiore a 7. Si controlli che l'interesse sia compreso tra 0 e 7. NOTA: se questo campo contiene il numero intero 2, significa che l'interesse è del 2%.

Tali dati vanno inseriti in un array di strutture `struct correntista` contenente questi campi.

Dopo la lettura dei dati, il programma deve aggiornare ogni cella dell'array con un nuovo importo calcolato come somma tra il vecchio importo e l'interesse maturato.

A tal fine si usi una funzione

```
float nuovoImportoCliente(struct correntista v);
```

Tale funzione prende una singola struttura correntista calcola il nuovo importo e lo fornisce in uscita

FACOLTATIVO:

Si organizzi il progetto su più file in modo da separare il main dalle funzioni e si crei il corrispondente file header.

NOTA: Si consegnino in questo caso TUTTI i FILE .c e il .h

```

#include <stdio.h>

struct correntista {char Cognome[21];
                    int NumeroConto;
                    float Importo;
                    int PercentualeInteresse;
                    };

float nuovoImportoCliente(struct correntista V)
{return V.Importo + V.Importo*(V.PercentualeInteresse/100.0);
}

void main()
{int i;
 struct correntista V[3];

 for (i=0; i<3; i++)
 {printf("Cognome del cliente #%d : ",i);
  scanf("%s", V[i].Cognome);
  printf("Numero conto del cliente #%d : ",i);
  scanf("%d", &V[i].NumeroConto);
  printf("Importo del cliente #%d : ",i);
  scanf("%f", &V[i].Importo);
  do
  {printf("Percentuale interesse del cliente #%d (0..7) : ",i);
   scanf("%d", &V[i].PercentualeInteresse);
  }
  while ((V[i].PercentualeInteresse>7) || (V[i].PercentualeInteresse<0));
 }

 for (i=0; i<3; i++)
  printf("Utente %s\tConto %d\tImporto %f\n", V[i].Cognome, V[i].NumeroConto,
        V[i].Importo=nuovoImportoCliente(V[i]));
}

```

PROVA PRATICA DI FONDAMENTI DI INFORMATICA L-A
Prof. MICHELA MILANO – 25 NOVEMBRE 2004
COMPITO E

Si scriva un programma C che permetta di inserire i dati relativi a uno spoglio elettorale. Per ogni candidato (si supponga per comodità che saranno inseriti al massimo 4 candidati) vengono inseriti dall'utente i seguenti dati

- **Cognome:** una stringa di lunghezza massima di 20 caratteri, non contenente spazi.
- **IdentificativoCandidato:** intero
- **VotiAnno2001:** intero (numero di voti presi su 100 votanti)
- **VotiAnno2004:** intero (numero di voti presi su 100 votanti). Si controlli che la differenza tra il numero dei voti dell'anno 2004 e il numero del 2001 sia compresa tra -10 e +10.

Tali dati vanno inseriti in un array di strutture `struct candidato`. Ogni struttura contiene questi campi.

Il programma, scorrendo l'array, deve stampare a terminale il cognome di ogni candidato e la differenza tra la percentuale di voti della corrente elezione (Anno 2004) e quella del 2001.

A tal fine si utilizzi una procedura:

```
void calcolascarto(struct candidato V[], int Dim);
```

Tale procedura, scorrendo l'array V di strutture candidato di dimensione Dim, effettua il conto richiesto e lo stampa.

FACOLTATIVO:

Si organizzi il progetto su più file in modo da separare il main dalle funzioni e si crei il corrispondente file header.

NOTA: Si consegnino in questo caso TUTTI i FILE .c e il .h

```

#include <stdio.h>

struct candidato {char Cognome [21];
                  int IdentificativoCandidato;
                  int VotiAnno2001;
                  int VotiAnno2004;
                  };

void calcolascarto(struct candidato V[],int Dim)
{int i;
  for(i=0;i<Dim;i++)
    printf("Candidato %s\tDifferenza %d\n",V[i].Cognome,V[i].
                                                VotiAnno2004-V[i].VotiAnno2001);
}

void main()
{struct candidato V[4];
  int i,n;
  do
    {printf("numero di candidati (max. 4): ");
      scanf("%d",&n);
    }
  while((n>4) || (n<0));
  for(i=0;i<n;i++)
    {printf("Cognome del candidato #%d : ",i);
      scanf("%s",V[i].Cognome);
      printf("Identificativo del candidato #%d : ",i);
      scanf("%d",&V[i].IdentificativoCandidato);
      do
        {printf("Voti anno 2001 del candidato #%d (max. 100) : ",i);
          scanf("%d",&V[i].VotiAnno2001);
        }
        while ((V[i].VotiAnno2001<0) || (V[i].VotiAnno2001>100));
      do
        {printf("Voti anno 2004 del candidato #%d (max.100) : ",i);
          scanf("%d",&V[i].VotiAnno2004);
        }
        while ((V[i].VotiAnno2004<0) || (V[i].VotiAnno2004>100) ||
              (V[i].VotiAnno2004>V[i].VotiAnno2001+10) ||
              (V[i].VotiAnno2004<V[i].VotiAnno2001-10));
    }

  calcolascarto(V,n);
}

```

PROVA PRATICA DI FONDAMENTI DI INFORMATICA L-A
Prof. MICHELA MILANO – 25 NOVEMBRE 2004
COMPITO F

Si scriva un programma C che permetta di inserire i dati relativi a uno spoglio elettorale. Per ogni candidato (si supponga per comodità che saranno inseriti al massimo 6 candidati) vengono inseriti dall'utente i seguenti dati

- **Cognome:** una stringa di lunghezza massima di 20 caratteri, non contenente spazi.
- **IdentificativoCandidato:** intero
- **VotiAnno2001:** intero (numero di voti presi su 100 votanti)
- **Previsione2004:** intero (scarto rispetto al 2001). Si controlli che questo numero, che rappresenta una percentuale, sia compreso tra -20 e +20. Esempio: se la previsione è -10 mi aspetto che questo candidato perda il 10% dei voti nel 2004, mentre se la previsione è +10 mi aspetto un incremento del 10% dei voti.

Tali dati vanno inseriti in un array di strutture `struct candidato`. Ogni struttura contiene questi campi.

Il programma, scorrendo l'array, deve stampare a terminale il cognome di ogni candidato e il relativo numero di voti attesi nell'Anno 2004.

A tal fine si utilizzi una procedura,

```
void stampaAttesi(struct candidato V[], int Dim);
```

Tale procedura, scorrendo l'array V di strutture candidato di dimensione Dim, effettua il conto richiesto e lo stampa.

FACOLTATIVO:

Si organizzi il progetto su più file in modo da separare il main dalle funzioni e si crei il corrispondente file header.

NOTA: Si consegnino in questo caso TUTTI i FILE .c e il .h

```

#include <stdio.h>

struct candidato {char Cognome [21];
                  int IdentificativoCandidato;
                  int VotiAnno2001;
                  int Previsione2004;
                  };

void stampaAttesi(struct candidato V[],int Dim)
{int i;
  for(i=0;i<Dim;i++)
    printf("Candidato %s\tPrevisione %f\n",V[i].Cognome,
          V[i].VotiAnno2001+V[i].VotiAnno2001*(V[i].Previsione2004/100.0));
}

void main()
{struct candidato V[6];
  int i,n;
  do
    {printf("numero di candidati (max. 6): ");
     scanf("%d",&n);
    }
  while((n>6) || (n<0));
  for(i=0;i<n;i++)
    {printf("Cognome del candidato #%d : ",i);
     scanf("%s",V[i].Cognome);
     printf("Identificativo del candidato #%d : ",i);
     scanf("%d",&V[i].IdentificativoCandidato);
     do
       {printf("Voti anno 2001 del candidato #%d (max. 100) : ",i);
        scanf("%d",&V[i].VotiAnno2001);
       }
     while ((V[i].VotiAnno2001<0) || (V[i].VotiAnno2001>100));
     do
       {printf("Voti anno 2004 del candidato #%d (max.100) : ",i);
        scanf("%d",&V[i].Previsione2004);
       }
     while ((V[i].Previsione2004<-20) || (V[i].Previsione2004>20));
    }

  stampaAttesi(V,n);
}

```