

1. Il prezzo della benzina è di euro 1.50 al litro. Scrivere una procedura che chiede all'utente quanti litri vuole comprare e stampa il prezzo corrispondente.

```
litri = raw_input("Quanti litri di benzina vuoi? ")
print "il costo è: ", float(litri) * 1.50
```

2. Scrivere una procedura che chiede all'utente tre numeri e stampa il loro prodotto.

```
x = raw_input("scrivi tre numeri")
y = raw_input()
z = raw_input()
print "il prodotto è: ", float(x)*float(y)*float(z)
```

3. Scrivere una procedura che svolge il seguente compito:
  - chiede a un piccolo imprenditore il nome dei suoi tre dipendenti e il numero di ore che hanno lavorato durante il giorno;
  - assumendo che la paga è di dieci euro l'ora, stampa il nome e la paga giornaliera di ognuno.

```
dip1 = raw_input("nome primo dipendente: ")
ore1 = raw_input("numero ore: ")
dip2 = raw_input("nome secondo dipendente: ")
ore2 = raw_input("numero ore: ")
dip3 = raw_input("nome terzo dipendente: ")
ore3 = raw_input("numero ore: ")
print dip1, 10*int(ore1), "euro"
print dip2, 10*int(ore2), "euro"
print dip3, 10*int(ore3), "euro"
```

4. Chiedi all'utente due numeri, se vuole invertire l'ordine di inserimento e stampa nell'ordine richiesto.

```
x = raw_input("scrivi due numeri")
y = raw_input()
z = raw_input("vuoi invertire l'ordine di inserimento? (se no inserisci 0, se si un altro intero)")
if int(z):
    print y, x
else:
    print x, y
```

5. Scrivere una procedura che chiede all'utente tre numeri e stampa il numero di valori distinti.

```
x = raw_input("scrivi tre numeri")
y = raw_input()
z = raw_input()
print "il numero di valori distinti è ",
if (x == y) and (y==z):
    print 1
elif (x == y) or (y==z) or (x==z):
    print 2
else:
    print 3
```

6. Scrivere una procedura che stampa il prezzo del biglietto di uno spettacolo in un club, secondo la regola: dieci euro se l'utente ha meno di sessanta anni e non è membro del club dove si tiene lo spettacolo, cinque se è un membro con meno di sessanta anni o, non essendo membro, ne ha almeno sessanta, tre, altrimenti.

```
età = raw_input("quanti anni hai? ")
membro = raw_input("sei un membro del club? (se no inserisci 0, se si un altro intero)")
print "il costo del biglietto è di euro ",
if (int(età) < 60) and not(int(membro)):
    print 10
elif (int(età) < 60) or not(int(membro)):
    print 5
else:
    print 3
```

7. Scrivere una procedura che legge i numeri dal file non vuoto NUMERI.pkl e stampa la loro somma.

```
import pickle
f = open("NUMERI.pkl", "r")
x = pickle.load(f)
somma = 0
while x != "":
    somma = somma + x
    x = pickle.load(f)
print somma
f.close()
```

8. Scrivere una procedura che legge i numeri dal file NUMERI.pkl e stampa la loro somma.

```
import pickle
f = open("NUMERI.pkl", "r")
x = pickle.load(f)
if x == "" :
    somma = ""
else:
    somma = 0
while x != "" :
    somma = somma + x
    x = pickle.load(f)
print somma
f.close()
```

9. Scrivere una procedura che legge i numeri dal file NUMERI.pkl e stampa il loro prodotto.

```
import pickle
f = open("NUMERI.pkl", "r")
x = pickle.load(f)
if x == "" :
    prodotto = ""
else:
    prodotto = 1
while x != "" :
    prodotto = prodotto * x
    x = pickle.load(f)
print prodotto
f.close()
```

10. Scrivere una procedura che legge i numeri dal file NUMERI.pkl e stampa il valore massimo.

```
import pickle
f = open("NUMERI.pkl", "r")
x = pickle.load(f)
massimo = x
while x != "":
    if massimo < x :
        massimo = x
    x = pickle.load(f)
print massimo
f.close()
```

11. Scrivere una procedura che legge i numeri dal file NUMERI.pkl e stampa il valore minimo.

Stessa soluzione dell' esercizio successivo, sostituendo l' operatore di confronto < con > e rinominando, opportunamente, la variabile di nome massimo in minimo.

12. Scrivere una procedura che legge i numeri dal file NUMERI.pkl e stampa la media dei valori.

```
import pickle
f = open("NUMERI.pkl", "r")
somma = 0
contatore=0
x = pickle.load(f)
while x != "":
    contatore = contatore + 1
    somma = somma + x
    x = pickle.load(f)
if contatore > 0:
    print somma/contatore
f.close()
```

13. Il file VIP.pkl contiene nome e anno di nascita di un insieme di personaggi deceduti. Chiedi all'utente di inserire l'anno di morte per ogni nome del file VIP.pkl in un file PERSONAGGI.pkl.

```
import pickle

f = open("VIP.pkl", "r")

g = open("PERSONAGGI.pkl", "w")

nome = pickle.load(f)

while nome != "":

    morte = raw_input("quale è l'anno di morte di " + nome + " ?")

    pickle.dump(nome, g)

    pickle.dump(pickle.load(f) , g)

    pickle.dump(morte, g)

    nome = pickle.load(f)

f.close()

g.close()
```

14. Il file ALUMNI.pkl contiene l'informazione su un gruppo di ex-studenti. Per ognuno, è memorizzato il nome e l'anno di laurea. Scrivere una procedura che stampa il nome e l'anno di laurea degli ex-studenti laureati nello scorso millennio in un file OLDALUMNI.PKL.

```
import pickle

f = open("ALUMNI.pkl", "r")

g = open("OLDALUMNI.pkl", "w")

nome= pickle.load(f)

while nome != "":

    anno_laurea = pickle.load(f)

    if anno_laurea < 2000 :

        pickle.dump(nome, g)

        pickle.dump(anno_laurea, g)

    nome = pickle.load(f)

f.close()

g.close()
```

15. Il file LAVORATORI.pkl contiene l'informazione su un gruppo di lavoratori. Per ognuno di essi, è memorizzato il nome e se è a tempo pieno (1 sì, 0 no). Scrivere una procedura che stampa il nome dei lavoratori a tempo pieno in un file TEMPOPIENO.PKL.

```
import pickle
f = open("LAVORATORI.pkl", "r")
g = open("TEMPOPIENO.pkl", "w")
nome= pickle.load(f)
while nome != "" :
    if pickle.load(f):
        pickle.dump(nome, g)
    nome = pickle.load(f)
f.close()
g.close()
```

16. Il file LAVORATORI.pkl contiene l'informazione su un gruppo di lavoratori. Per ognuno di essi, è memorizzato il nome e l'anno di assunzione. Scrivere una procedura che stampa il nome e anno di assunzione dei lavoratori che sono almeno al decimo anno di servizio in un file PROMOVBILI.pkl.

```
import pickle
f = open("LAVORATORI.pkl", "r")
g = open("PROMOVIBILI.pkl", "w")
anno_corrente = raw_input("inserisci anno corrente")
anno_corrente = int(anno_corrente)
nome= pickle.load(f)
while nome != "" :
    anno_assunzione = pickle.load(f)
    if anno_corrente - anno_assunzione >= 10 :
        pickle.dump(nome, g)
        pickle.dump(anno_assunzione, g)
    nome = pickle.load(f)
f.close()
g.close()
```

17. Scrivere una procedura che copia in una lista i numeri dal file NUMERI.pkl.

```
import pickle
f = open("NUMERI.pkl", "r")
lista = []
x = pickle.load(f)
while x != "":
    lista = lista + [x]
    x = pickle.load(f)
f.close()
```

18. Il file LAVORATORI.pkl contiene l'informazione su un gruppo di lavoratori. Per ognuno, è memorizzato il nome e l'anno di assunzione. Scrivere una procedura che stampa il nome dei lavoratori assunti nell'anno di assunzione più remoto in un file PENSIONABILI.pkl.

```
import pickle
class Lavoratore:
    pass
f = open("LAVORATORI.pkl", "r")
g = open("PENSIONABILI.pkl", "w")
lavoratore = Lavoratore()
lavoratore.nome = pickle.load(f)
lavoratore.assunzione = pickle.load(f)
minimo = lavoratore.assunzione
lavoratori = []
while lavoratore.nome != "":
    if lavoratore.assunzione < minimo:
        minimo = lavoratore.assunzione
    lavoratori = lavoratori + [lavoratore]
    lavoratore.nome = pickle.load(f)
    lavoratore.assunzione = pickle.load(f)
for lavoratore in lavoratori:
    if lavoratore.assunzione == minimo:
        pickle.dump(lavoratore, g)
f.close()
g.close()
```