

Esercizi

Programmazione I e Informatica II

9 ottobre 2015

Esercizio 1 Hello World

Stampare la stringa “Hello World!” sullo schermo.

Soluzione dell’esercizio 1

```
1 #include <stdio.h>
2
3 /* My first program */
4 int main(int argc, char *argv[])
5 {
6     printf("Hello world.");
7
8     return 0;
9 }
```

Esercizio 2 Variabili

Dichiarare una variabile *char*, due variabili *int*, una variabile *float* (nomi a scelta), e poi assegnarle rispettivamente i valori ‘e’, 10, 15, e 12.6. Stampare il nome di ciascuna variabile ed il suo valore sull schermo. Utilizzare %c, %d, %f per formattare l’output.

Soluzione dell’esercizio 2

```
1 #include <stdio.h>
2
3 int main(int argc, char *argv[])
4 {
5     char a;
6     int x;
7     int y;
8     float z;
9     a = 'e';
10    x=10;
11    y=15;
12    z=12.6;
13
14    printf("a= %c",a);
15    printf("\n");
}
```

```

16 printf("x= %d",x);
17 printf("\t");
18 printf("y= %d",y);
19 printf("\t");
20 printf("z= %f",z);
21 printf("\n");
22
23 return 0;
24 }

```

Esercizio 3 Moltiplicazione di due float

Definire due variabili con nome *a* e *b* entrambe di tipo *float* e poi stamparle su schermo (prima di stamparle, *a* ha valore 5.678 e *b* ha valore 1.23456). Poi definire una terza variabile di tipo *double* (nome a scelta dello studente) e assegnarle il valore derivante dalla moltiplicazione di *a* * *b*. Infine, stampare questa terza variabile.

Soluzione dell'esercizio 3

```

1 #include <stdio.h>
2
3 int main(int argc, char *argv[])
4 {
5     float a = 5.678;
6     float b = 1.23456;
7
8     printf("Stampa di a: %f\n", a);
9     printf("Stampa di b: %f\n", b);
10
11     double c = a*b;
12     printf("Stampa di b: %f\n", c);
13
14
15     return 0;
16 }

```

Esercizio 4 Enum giorni settimana

Definire un tipo *enum weekdays* che contiene i giorni della settimana a partire da *sunday* (il cui valore corrisponde a 0 quindi). Definire due variabili *today* e *tomorrow* di tipo *enum weekdays*. Assegnare a *today* il giorno di oggi (*friday*), e a *tomorrow* il giorno successivo (farlo utilizzando un'operazione aritmetica, utilizzando *today* + 1). Stampare il valore intero corrispondente a *tomorrow*. Formato di stampa %d.

In secondo luogo, sempre all'interno della stessa funzione *main()*, definire anche un *enum* senza nome che continene solo *eastern* (Pasqua), che deve avere valore uguale al valore di *sunday* in *enum weekdays* (cioè 0). Alla fine della solita funzione *main()*, controllare che *today* non sia *eastern* (valore differente): in pratica, controllare che oggi non possa essere Pasqua! Questo viene controllato comparando i due valori con un comando condizionale *if*. Quindi, stampare "Oggi non puo' essere Pasqua\n", in caso *if(pasqua != today)*.

Soluzione dell'esercizio 4

```
1 #include <stdio.h>
2
3 int main(int argc, char *argv[])
4 {
5     enum week{sunday, monday, tuesday, wednesday, thursday, friday,
6         saturday};
7
8     //Esempio di tipo enum senza nome che mi serve serve per
9     //definire una costante eastern (ha valore equivalente a sunday)
10    enum {eastern = 0};
11
12    enum week today, tomorrow;
13    today= friday;
14    tomorrow = today + 1;
15    printf("Tomorrow is %d\n", tomorrow);
16
17    //Scommentare se volete stampare il nome del giorno
18    /*
19    if (today == sunday)
20        printf("Today is Sunday\n");
21    else
22        if (today == monday)
23            printf("Today is Monday\n");
24        else
25            if (today == tuesday)
26                printf("Today is Tuesday\n");
27            else
28                if (today == wednesday)
29                    printf("Today is Wednesday\n");
30                else
31                    if (today == thursday)
32                        printf("Today is Thursday\n");
33                    else
34                        if (today == friday)
35                            printf("Today is Friday\n");
36                        else
37                            if (today == saturday)
38                                printf("Today is Saturday\n");
39    */
40
41    // Se oggi non e' domenica, non puo' essere Pasqua!
42    if (pasqua != today)
43        printf("Today cannot not be Eastern\n");
44
45    return 0;
46 }
```

Esercizio 5 Funzione valore assoluto

Il file .c di questo esercizio deve contenere nell'ordine, il prototipo di una funzione *abs*, che prende in input un intero e restituisce un intero (*int*). Sotto ad esso nel file .c, la funzione *main*, dove viene definita una variabile di nome *pippo* a cui viene assegnato il valore -3. La funzione *main* chiama la funzione *abs* passandole come parametro *pippo*, e poi infine stampa il risultato ottenuto

dalla funzione *abs*. Infine il file *.c* contiene la definizione della funzione *abs*; tale funzione calcola il valore assoluto del parametro passato e lo ritorna al chiamante (in questo caso la funzione *main()*).

- Domanda: se definisco una variabile *int pippo = -7* prima del prototipo di *abs*, quindi il relativo identificatore ha visibilit  di file (*file scope*), cosa viene stampato in *main()*? 3 o 7?

Soluzione dell'esercizio 5

```
1 #include <stdio.h>
2
3 //Prototipo di funzione
4 int absValue (int);
5
6
7 //Funzione main()
8 int main(int argc, char *argv[])
9 {
10     int pippo= -3;
11     int result= absValue(pippo);
12
13     printf("The absolute value of %d is %d.\n", pippo, result);
14 }
15
16 //Definizione della funzione absvalue()
17 int absValue (int value)
18 {
19
20     int returnValue= value;
21
22     if (returnValue < 0)
23         returnValue = -returnValue;
24
25     return returnValue;
26 }
27
```

Esercizio 6 Numeri in virgola mobile

Supponendo di leggere i seguenti bit da sinistra verso destra (il bit meno significativo e' il primo), i 2byte di questo esercizio mi rappresentano un numero in virgola mobile (*floating point*). Il primo bit rappresenta il segno (*s*), i secondi 4 bit rappresentano l'esponente (*q*). I rimanenti 11 bit rappresentano la mantissa *c* (o significando). In questo esercizio l'esponente *q* e' rappresentato in modulo a segno (se il suo bit piu' significativo e' 0 ha segno positivo, se 1 ha segno negativo). Svolgere l'esercizio con carta e penna, non e' un esercizio di compilazione. Detto questo,

- calcolare il valore in base 10 dalla stringa 1, utilizzando la formula $(-1)^s \times c \times 2^q$.
- Eseguire lo stesso calcolo per la stringa 2.

$$1|1000|00110000000 \quad (1)$$

$$0|1001|00010000000 \quad (2)$$

Soluzione dell'esercizio 6

$$(1) \quad (-1)^1 \times 12 \times 2^1 = -24$$

$$(2) \quad (-1)^0 \times 8 \times 2^{-1} = 4$$