

A.A. 08/09

Fondamenti di Programmazione

(canale E-O)

Docente: Prof.ssa Tiziana Calamoneri
calamo@di.uniroma1.it

Esercitatore: Dott. Roberto Petroccia
petroccia@di.uniroma1.it

Pagina del corso:

<http://twiki.di.uniroma1.it/twiki/view/Programmazione1/EO/WebHome>

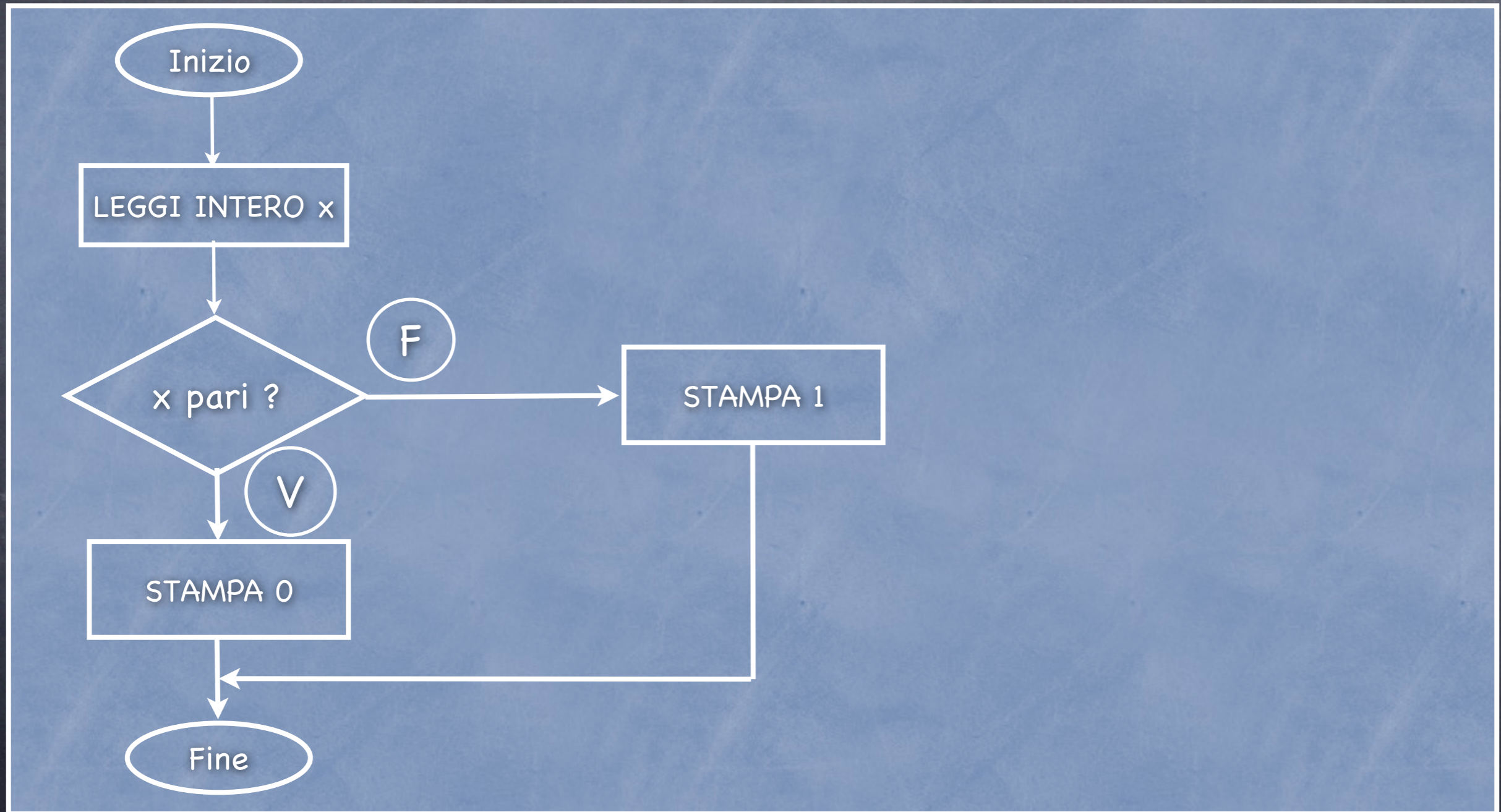
Esercitazione del 08/10/08

Soluzioni esercizi

1) Si legga un intero e si stampi 1 se questo è dispari, e 0 se è pari.

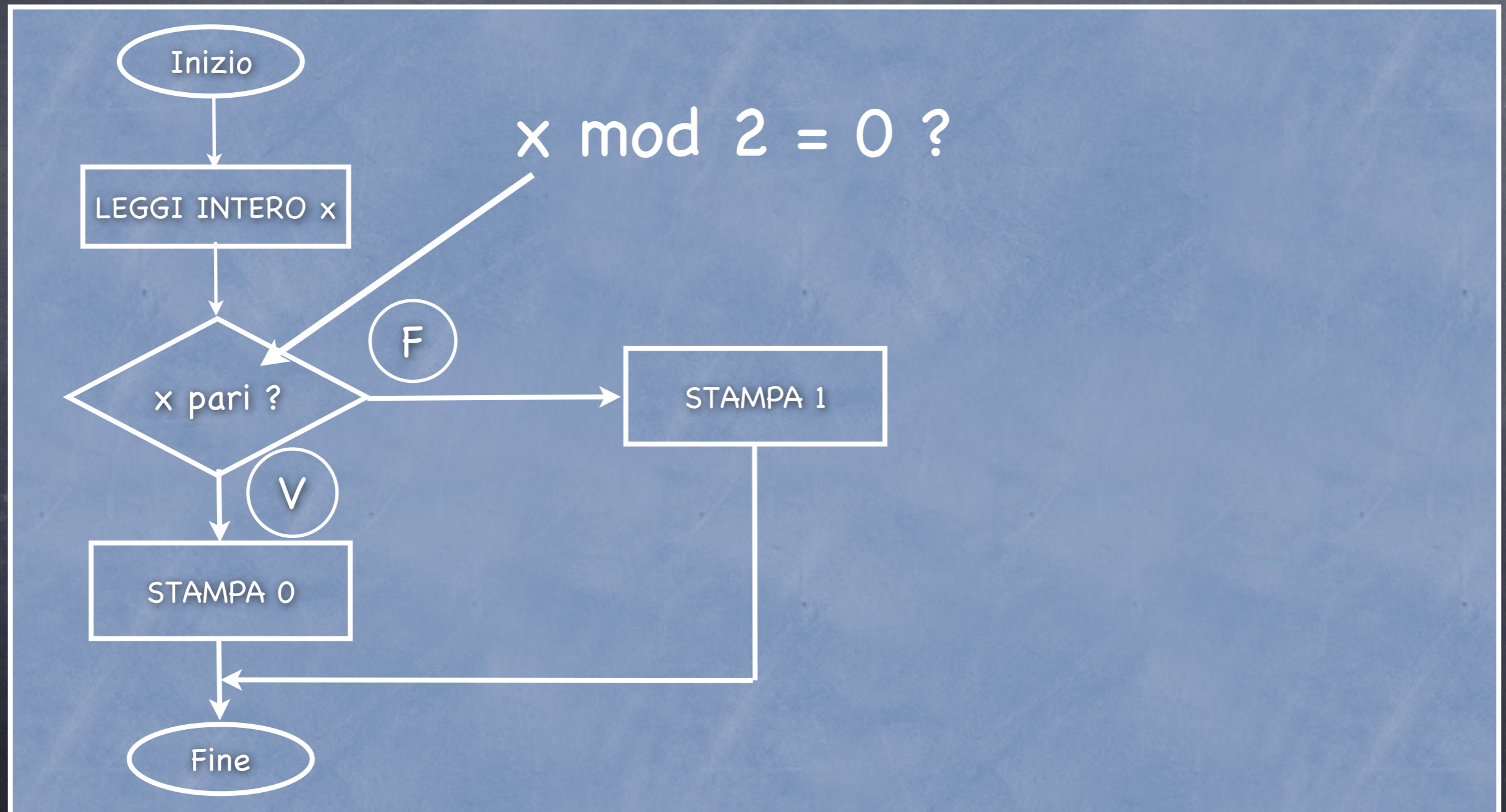
Soluzioni esercizi

1) Si legga un intero e si stampi 1 se questo è dispari, e 0 se è pari.



Soluzioni esercizi

1) Si legga un intero e si stampi 1 se questo è dispari, e 0 se è pari.



Soluzioni esercizi

1) Si legga un intero e si stampi 1 se questo è dispari, e 0 se è pari.

```
#include <stdio.h>
int main() {
    int x = 0;
    printf ("Inserisci un intero:\n");
    scanf ("%d", &x);
    if (x % 2 == 0) { // x e' pari
        printf("0\n");
    }
    else { // x e' dispari
        printf("1\n");
    }
    return 0;
}
```

Soluzioni esercizi

1) Si legga un intero e si stampi 1 se questo è dispari, e 0 se è pari.

```
#include <stdio.h>
int main() {
    int x = 0;
    printf ("Inserisci un intero:\n");
    scanf ("%d, &x);
    if (x % 2 == 0) { // x e' pari
        printf("0\n");
    }
    else { // x e' dispari
        printf("1\n");
    }
    return 0;
}
```

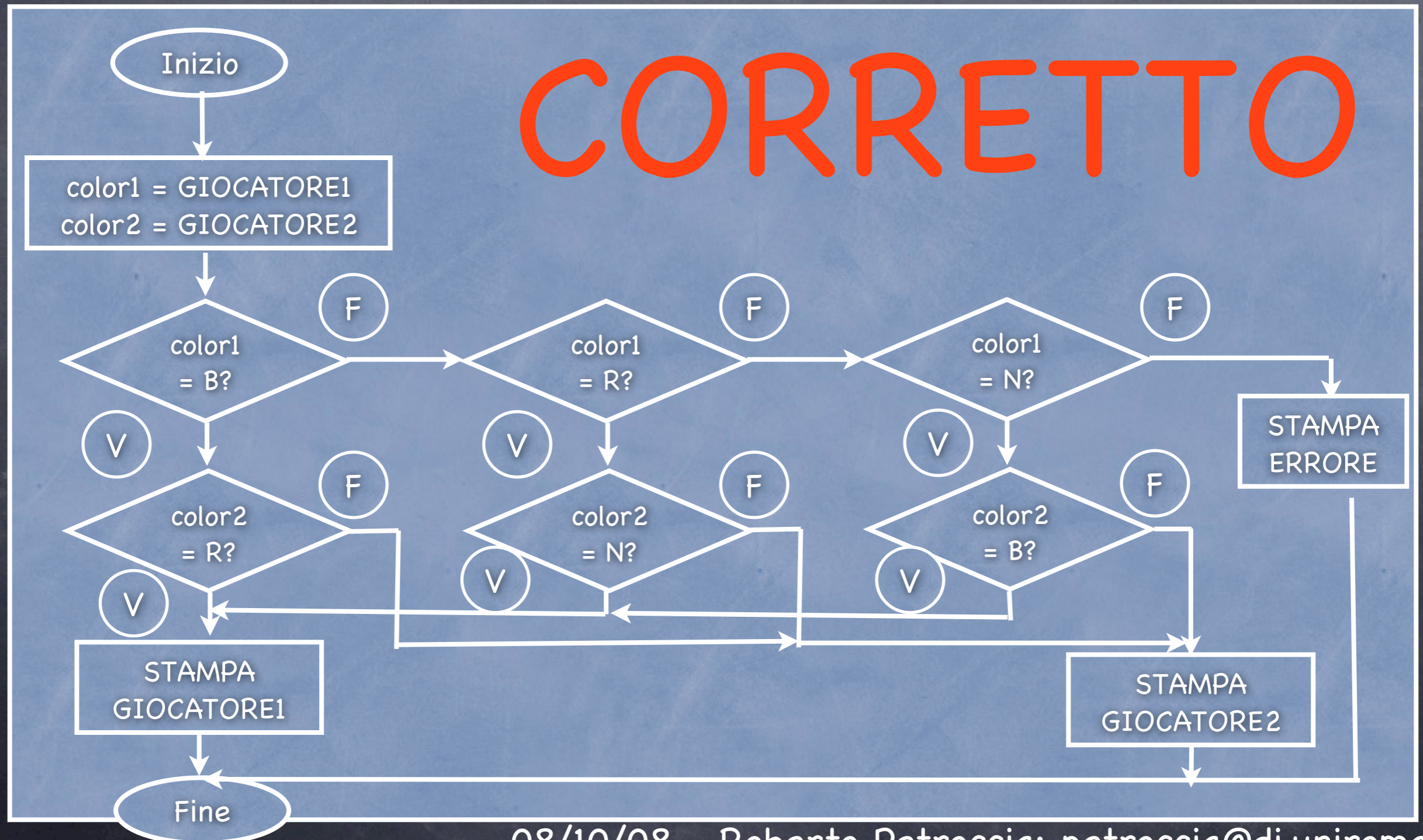
```
#include <stdio.h>
int main() {
    int x = 0;
    printf ("Inserisci un intero:\n");
    scanf ("%d, &x);
    if (x % 2 == 0) { printf("0\n"); }
    else { printf("1\n");}
    return 0;
}
```

Soluzioni esercizi

- 3) Ci sono 2 giocatori. Ogni giocatore sceglie un colore tra bianco (B), rosso (R), nero (N). Il bianco batte il rosso, il rosso batte il nero, il nero batte il bianco. Stampare il vincitore.

Soluzioni esercizi

- 3) Ci sono 2 giocatori. Ogni giocatore sceglie un colore tra bianco (B), rosso (R), nero (N). Il bianco batte il rosso, il rosso batte il nero, il nero batte il bianco. Stampare il vincitore.



Soluzioni esercizi

3) Ci sono 2 giocatori. Ogni giocatore sceglie un colore tra bianco (B), rosso (R), nero (N). Il bianco batte il rosso, il rosso batte il nero, il nero batte il bianco. Stampare il vincitore.

Considera color1 per colore del Giocatore1

Considera color2 per colore del Giocatore2

LEGGI colore giocatore1

LEGGI colore giocatore2

Controlla color1

Controlla color2

se color1 è bianco e color2 è rosso

 STAMPA Giocatore1

 esci

se color1 è rosso e color2 è nero

 STAMPA Giocatore1

 esci

se color1 è nero e color2 è bianco

 STAMPA Giocatore1

 esci

STAMPA Giocatore1

esci

Assumiamo che in caso di
pareggio il Giocatore2 vinca

Soluzioni esercizi

- 3) Ci sono 2 giocatori. Ogni giocatore sceglie un colore tra bianco (B), rosso (R), nero (N). Il bianco batte il rosso, il rosso batte il nero, il nero batte il bianco. Stampare il vincitore.

```
#include <stdio.h>
#define B 0
#define R 1
#define N 2
int main() {
    int color1 = -1;
    int color2 = -1;
    printf ("Inserisci colore giocatore 1:\n");
    scanf ("%d, & color1);
    if (color1 < 0 || color1 > 2) {
        printf("Il giocatore 1 ha selezionato un colore sbagliato\n");
        return -1;
    }
    ...
}
```

Soluzioni esercizi

3) Ci sono 2 giocatori. Ogni giocatore sceglie un colore tra bianco (B), rosso (R), nero (N). Il bianco batte il rosso, il rosso batte il nero, il nero batte il bianco. Stampare il vincitore.

```
...
printf ("Inserisci colore giocatore 2:\n");
scanf ("%d, & color2);
if (color2 < 0 || color2 > 2) {
    printf("Il giocatore 2 ha selezionato un colore sbagliato\n");
    return -1;
}
if (color1 == B && color2 == R) {
    printf("Giocatore1\n");
    return 0;
}
...
```

Soluzioni esercizi

3) Ci sono 2 giocatori. Ogni giocatore sceglie un colore tra bianco (B), rosso (R), nero (N). Il bianco batte il rosso, il rosso batte il nero, il nero batte il bianco. Stampare il vincitore.

```
...
if (color1 == R && color2 == N) {
    printf("Giocatore1\n");
    return 0;
}
if (color1 == N && color2 == B) {
    printf("Giocatore1\n");
    return 0;
}

printf("Giocatore2\n"); // hanno scelto stesso colore
return 0;
}
```

Operatori aritmetici

Siano a , b e c degli interi che valgono, rispettivamente, 8, 3 e -5. Determinare il valore delle seguenti espressioni:

- 1) $a + b + c$
- 2) $a * b + c$
- 3) $a * (b + c)$
- 4) $(a / b) * c$

```
#include <stdio.h>
#define a 8
#define b 3
#define c -5
int main() {
    printf("a + b + c = %d\n", a + b + c); // risultato e' 6
    printf("a * b + c = %d\n", a * b + c); // risultato e' 19
    printf("a * (b + c) = %d\n", a * (b + c)); // risultato e' -16
    printf("(a / b) * c = %d\n", (a / b) * c); // risultato e' -10 (8/3 = 2,6666)
    printf("(a / b) * c = %f\n", (a / b) * c); // risultato e' 0.0000 perchè?
    return 0;
}
```

Operatori aritmetici

Siano a , b e c degli interi che valgono, rispettivamente, 8, 3 e -5. Determinare il valore delle seguenti espressioni:

- 1) $a + b + c$
- 2) $a * b + c$
- 3) $a * (b + c)$
- 4) $(a / b) * c$

```
#include <stdio.h>
#define a 8.0
#define b 3.0
#define c -5.0
int main() {
    //(8/3 = 2,6666)
    printf("(a / b) * c = %d\n", (a / b) * c); // risultato e' qualcosa di strano
    printf("(a / b) * c = %f\n", (a / b) * c); // risultato e' -13,333
    printf("(a / b) * c = %f\n", (int)((a / b) * c)); // risultato e' -13
    return 0;
}
```

Assegnamento

Valutare i valori delle variabili in gioco alla fine dei seguenti frammenti di codice con $x = 3$ e $y = 2$

1) $y += x++$

2) $y *= ++x$

3) $x += ++x - y--$

Assegnamento

Valutare i valori delle variabili in gioco alla fine dei seguenti frammenti di codice con $x = 3$ e $y = 2$

1) $y += x++$

```
#include <stdio.h>
int main() {
    int x = 3, y = 2;
    y += x++;
    printf("y = %d x = %d\n", y, x); // y = 5 x = 4
    x = 3, y = 2;
```

2) $y *= ++x$

```
y *= ++x;
    printf("y = %d x = %d\n", y, x); // y = 8 x = 4
    x = 3, y = 2;
```

3) $x += ++x - y--$

```
x += ++x - y--;
    printf("y = %d x = %d\n", y, x); // y = 1 x = 6
    x = 3, y = 2;
    int z = 3;
z += ++x - y--;
    printf("y = %d x = %d z = %d\n", y, x, z); // y = 1 x = 4 z = 5
    return 0;
}
```


Esercizi

1) Scrivere un programma che letti due cateti restituisca il valore dell'ipotenusa

LEGGI cateto1

LEGGI cateto2

$$\text{ipotenusa} = \sqrt{(\text{cateto1} * \text{cateto1}) + (\text{cateto2} * \text{cateto2})}$$

STAMPA ipotenusa

Esercizi

1) Scrivere un programma che letti due cateti restituisca il valore dell'ipotenusa

LEGGI cateto1

LEGGI cateto2

$$\text{ipotenusa} = \sqrt{(\text{cateto1} * \text{cateto1}) + (\text{cateto2} * \text{cateto2})}$$

STAMPA ipotenusa

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

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

```
int main() {
```

```
    float cateto1, cateto2;
```

```
    scanf("%f", &cateto1);
```

```
    scanf("%f", &cateto2);
```

```
    float ipotenusa = sqrt((cateto1 * cateto1) + cateto2 * cateto2); //sqrt calcola la radice quadrata
```

```
    printf("ipotenusa = %f\n", ipotenusa);
```

```
    return 0;
```

```
}
```

Esercizi

2) Visualizzare gli interi pari da 2 a 100

*Considera un contatore i
 $i = 2$
finche' (i e' ≤ 100)
se i e' pari stampa i
incrementa i
esci*

Esercizi

2) Visualizzare gli interi pari da 2 a 100

*Considera un contatore i
 $i = 2$
finche' ($i \leq 100$)
se i e' pari stampa i
incrementa i
esci*

```
#include <stdio.h>
#define MIN_VALUE 2
#define MAX_VALUE 100
int main() {
    int i;
    for (i = MIN_VALUE; i <= MAX_VALUE; i++) {
        // inizializza la variabile, controlla la condizione ed alla
        // fine dell'iterazione corrente aggiorna il valore del contatore
        if (i % 2 == 0) {
            printf("%d\n", i);
        }
    }
    return 0;
}
```

Esercizi

2) Visualizzare gli interi pari da 2 a 100

*Considera un contatore i
 $i = 2$
finche' (i e' ≤ 100)
se i e' pari stampa i
incrementa i
esci*

```
#include <stdio.h>
#define MIN_VALUE 2
#define MAX_VALUE 100
int main() {
    int i = MIN_VALUE;
    while (i <= MAX_VALUE) { //controlla la condizione se e' vera entra nel ciclo
        if (i % 2 == 0) {
            printf("%d\n", i);
        }
        i++; // bisogna ricordarsi di incrementare la variabile o aggiornare lo stato della condizione
    }
    return 0;
}
```

Esercizi

2) Visualizzare gli interi pari da 2 a 100

*Considera un contatore i
 $i = 2$
finche' ($i \leq 100$)
se i e' pari stampa i
incrementa i
esci*

```
#include <stdio.h>
#define MIN_VALUE 2
#define MAX_VALUE 100
int main() {
    int i = MIN_VALUE;
    do { //la prima volta entra nel ciclo senza controllare la condizione
        if (i % 2 == 0) {
            printf("%d\n", i);
        }
        i++; // bisogna ricordarsi di incrementare la variabile o aggiornare lo stato della condizione
    }
    while (i <= MAX_VALUE);
    return 0;
}
```

Esercizi

3) Stampa un rettangolo di 10 righe e 20 colonne di *

Esercizi

3) Stampa un rettangolo di 10 righe e 20 colonne di *

Considera un contatore i

$i = 0$

finche' ($i \leq 10$)

*stampa 20 simboli * e vai a capo*

incrementa i

esci

Esercizi

3) Stampa un rettangolo di 10 righe e 20 colonne di *

Considera un contatore i

$i = 0$

finche' ($i < 10$)

*stampa 20 simboli * e vai a capo*

incrementa i

esci

```
#include <stdio.h>
#define NUM_RIGHE 10
#define NUM_COLONNE 20
int main() {
    int i = j = 0;
    for (i = 0; i < NUM_RIGHE; i++) {
        for (j = 0; j < NUM_COLONNE; j++) {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

Esercizi

4) Stampa un quadrato composto da tutte A con dimensione n presa in input

Esercizi

4) Stampa un quadrato composto da tutte A con dimensione n presa in input

Leggi n

Considera un contatore i

i = 0

finche' (i e' < n)

stampa n caratteri A e vai a capo

incrementa i

esci

Esercizi

4) Stampa un quadrato composto da tutte A con dimensione n presa in input

Leggi n

Considera un contatore i

i = 0

finche' (i e' < n)

stampa n caratteri A e vai a capo

incrementa i

esci

```
#include <stdio.h>
int main() {
    int n = 0;
    printf("Inserisci il valore n \n");
    scanf("%d", &n);
    int i = j = 0;
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            printf("A");
        }
        printf("\n");
    }
    return 0;
}
```

Esercizi

5) Preso un intero $n > 0$ in input stampare i primi 10 multipli di n
Esempio: $n = 2$, stampare 2, 4, 6, 8, 10, 12, 14, 16, 18, 20

Esercizi

5) Preso un intero $n > 0$ in input stampare i primi 10 multipli di n

Esempio: $n = 2$, stampare 2, 4, 6, 8, 10, 12, 14, 16, 18, 20

Leggi n

Considera un contatore i

$i = 0$

finche' ($i \leq 10$)

*Stampa $n * i$*

incrementa i

esci

Esercizi

5) Preso un intero $n > 0$ in input stampare i primi 10 multipli di n

Esempio: $n = 2$, stampare 2, 4, 6, 8, 10, 12, 14, 16, 18, 20

Leggi n

Considera un contatore i

$i = 0$

finche' ($i \leq 10$)

*Stampa $n * i$*

incrementa i

esci

```
#include <stdio.h>
#define MULTIPLI 10
int main() {
    int n = 0;
    printf("Inserisci il valore n \n");
    scanf("%d", &n);
    int i = 1;
    for (i = 1; i <= MULTIPLI; i++) {
        printf("%d\n", i * n);
    }
    return 0;
}
```

Esercizi

5) Preso un intero $n > 0$ in input stampare i primi 10 multipli di n

Esempio: $n = 2$, stampare 2, 4, 6, 8, 10, 12, 14, 16, 18, 20

Leggi n

Considera un contatore i

$i = 0$

finche' ($i \leq 10$)

*Stampa $n * i$*

incrementa i

esci

```
#include <stdio.h>
#define MULTIPLI 10
int main() {
    int n = 0;
    printf("Inserisci il valore n \n");
    scanf("%d", &n);
    int i = 1;
    for (i = 1; i <= MULTIPLI; i++) {
        printf("%d\n", i * n);
    }
    return 0;
}
```


switch

5) Prendere in input un intero tra da 1 a 12 e stampare il mese corrispondente (3 = Marzo)

switch

5) Prendere in input un intero tra da 1 a 12 e stampare il mese corrispondente (3 = Marzo)

Leggi n

Controlla se n e' corretto

Stampa mese corrispondente

esci

switch

5) Prendere in input un intero tra da 1 a 12 e stampare il mese corrispondente (3 = Marzo)

```
#include <stdio.h>
int main() {
    int n = 0;
    printf("Inserisci il mese n \n");
    scanf("%d", &n);
    switch (n) {
        case 1:
            printf("Gennaio\n");
            break;
            .....
        case 12:
            printf("Dicembre\n");
            break;
        default:
            printf("Il valore inserito non e' corretto\n");
    }
    return 0;
}
```

Leggi n

Controlla se n e' corretto

Stampa mese corrispondente

esci

Se inseriamo 3 abbiamo in output: Marzo

switch

5) Prendere in input un intero tra da 1 a 12 e stampare il mese corrispondente (3 = Marzo)

```
#include <stdio.h>
int main() {
    int n = 0;
    printf("Inserisci il mese n \n");
    scanf("%d", &n);
    switch (n) {
        case 1:
            printf("Gennaio\n");
            break;
            .....
        case 12:
            printf("Dicembre\n");
            break;
        default:
            printf("Il valore inserito non e' corretto\n");
    }
    return 0;
}
```

Leggi n

Controlla se n e' corretto

Stampa mese corrispondente

esci

Cosa succede se scordiamo i break



switch

5) Prendere in input un intero tra da 1 a 12 e stampare il mese corrispondente (3 = Marzo)

```
#include <stdio.h>
int main() {
    int n = 0;
    printf("Inserisci il mese n \n");
    scanf("%d", &n);
    switch (n) {
        case 1:
            printf("Gennaio\n");
            .....
        case 12:
            printf("Dicembre\n");
        default:
            printf("Il valore inserito non e' corretto\n");
    }
    return 0;
}
```

Leggi n

Controlla se n e' corretto

Stampa mese corrispondente

esci

Se inseriamo 3 abbiamo in output:

Marzo

Aprile

Maggio

Giugno

Luglio

...

Dicembre

Il valore inserito non e' corretto

DOMANDE ???