

# Inizializzare\_gioco()

## Dati di input

Nome	Descrizione	Vincoli
Record_file	file sul quale ci sono i record dei giocatori	Binario [esiste,non esiste]
Partite_salvate	file sul quale ci sono le partite salvate	Binario [esiste,non esiste]
DEFAULT	costante che contiene il numero di giocatori base	intero 2
Partite_salvate	2 casi di equivalenza	[esiste,non esiste]
Record_file	2 classi di equivalenza	[esiste,non esiste]
DEFAULT	1 classe di equivalenza	

TOTALE CASI DI TEST:  $2*2*1=4$

## Casi di test

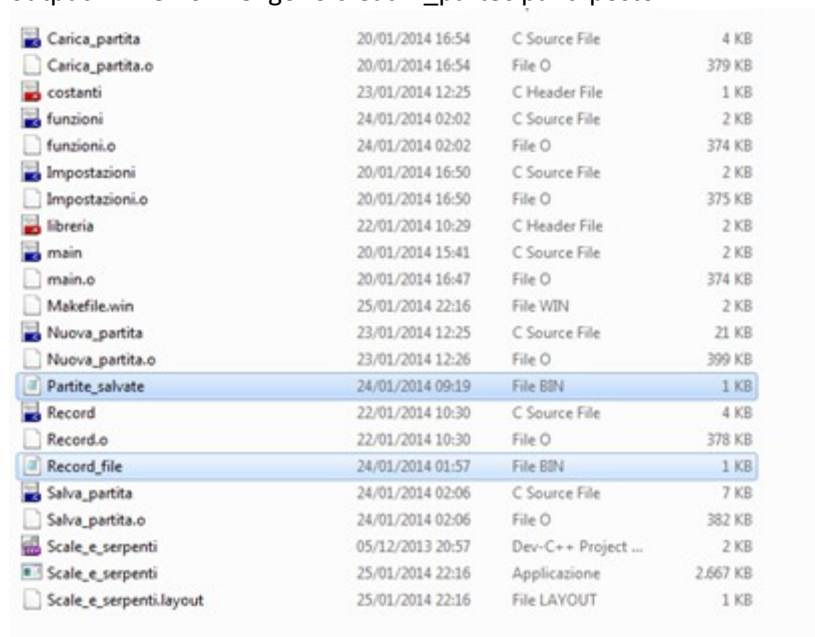
### Caso 1:

Record\_file = esiste

Partite\_salvate=esiste

DEFAULT=2

output " i file non vengono creati n\_partecipanti posto = 2"



Carica_partita	20/01/2014 16:54	C Source File	4 KB
Carica_partita.o	20/01/2014 16:54	File O	379 KB
costanti	23/01/2014 12:25	C Header File	1 KB
funzioni	24/01/2014 02:02	C Source File	2 KB
funzioni.o	24/01/2014 02:02	File O	374 KB
Impostazioni	20/01/2014 16:50	C Source File	2 KB
Impostazioni.o	20/01/2014 16:50	File O	375 KB
libreria	22/01/2014 10:29	C Header File	2 KB
main	20/01/2014 15:41	C Source File	2 KB
main.o	20/01/2014 16:47	File O	374 KB
Makefile.win	25/01/2014 22:16	File WIN	2 KB
Nuova_partita	23/01/2014 12:25	C Source File	21 KB
Nuova_partita.o	23/01/2014 12:26	File O	399 KB
Partite_salvate	24/01/2014 09:19	File BBN	1 KB
Record	22/01/2014 10:30	C Source File	4 KB
Record.o	22/01/2014 10:30	File O	378 KB
Record_file	24/01/2014 01:57	File BBN	1 KB
Salva_partita	24/01/2014 02:06	C Source File	7 KB
Salva_partita.o	24/01/2014 02:06	File O	382 KB
Scale_e_serpenti	05/12/2013 20:57	Dev-C++ Project ...	2 KB
Scale_e_serpenti	25/01/2014 22:16	Applicazione	2.667 KB
Scale_e_serpenti.layout	25/01/2014 22:16	File LAYOUT	1 KB

### Caso 2

Record\_file=non esiste

Partite\_salvate=non esiste

DEFAULT=2

output "i file vengono creati vuoti n\_partecipanti posto = 2"

Carica_partita	20/01/2014 16:54	C Source File	4 KB
Carica_partita.o	20/01/2014 16:54	File O	379 KB
costanti	23/01/2014 12:25	C Header File	1 KB
funzioni	24/01/2014 02:02	C Source File	2 KB
funzioni.o	24/01/2014 02:02	File O	374 KB
Impostazioni	20/01/2014 16:50	C Source File	2 KB
Impostazioni.o	20/01/2014 16:50	File O	375 KB
libreria	22/01/2014 10:29	C Header File	2 KB
main	20/01/2014 15:41	C Source File	2 KB
main.o	20/01/2014 16:47	File O	374 KB
Makefile.win	25/01/2014 22:16	File WIN	2 KB
Nuova_partita	23/01/2014 12:25	C Source File	21 KB
Nuova_partita.o	23/01/2014 12:26	File O	399 KB
Partite_salvate	25/01/2014 22:20	File BIN	0 KB
Record	22/01/2014 10:30	C Source File	4 KB
Record.o	22/01/2014 10:30	File O	378 KB
Record_file	25/01/2014 22:20	File BIN	1 KB
Salva_partita	24/01/2014 02:06	C Source File	7 KB
Salva_partita.o	24/01/2014 02:06	File O	382 KB
Scale_e_serpenti	05/12/2013 20:57	Dev-C++ Project ...	2 KB
Scale_e_serpenti	25/01/2014 22:16	Applicazione	2.667 KB
Scale_e_serpenti.layout	25/01/2014 22:16	File LAYOUT	1 KB

### Caso 3

Record\_file=non esiste

Partite\_salvate=esiste

DEFAULT =2

output " viene creato Record\_file ma vuoto e n\_partecipanti posto = 2"

Carica_partita	20/01/2014 16:54	C Source File	4 KB
Carica_partita.o	20/01/2014 16:54	File O	379 KB
costanti	23/01/2014 12:25	C Header File	1 KB
funzioni	24/01/2014 02:02	C Source File	2 KB
funzioni.o	24/01/2014 02:02	File O	374 KB
Impostazioni	20/01/2014 16:50	C Source File	2 KB
Impostazioni.o	20/01/2014 16:50	File O	375 KB
libreria	22/01/2014 10:29	C Header File	2 KB
main	20/01/2014 15:41	C Source File	2 KB
main.o	20/01/2014 16:47	File O	374 KB
Makefile.win	25/01/2014 22:16	File WIN	2 KB
Nuova_partita	23/01/2014 12:25	C Source File	21 KB
Nuova_partita.o	23/01/2014 12:26	File O	399 KB
Partite_salvate	24/01/2014 09:19	File BIN	1 KB
Record	22/01/2014 10:30	C Source File	4 KB
Record.o	22/01/2014 10:30	File O	378 KB
Record_file	24/01/2014 01:57	File BIN	1 KB
Salva_partita	24/01/2014 02:06	C Source File	7 KB
Salva_partita.o	24/01/2014 02:06	File O	382 KB
Scale_e_serpenti	05/12/2013 20:57	Dev-C++ Project ...	2 KB
Scale_e_serpenti	25/01/2014 22:16	Applicazione	2.667 KB
Scale_e_serpenti.layout	25/01/2014 22:16	File LAYOUT	1 KB

### Caso 4

Record\_file=non esiste

Partite\_salvate=non esiste

DEFAULT =2

output "viene creato Partite\_salvate ma vuoto n\_partecipanti posto = 2"

Carica_partita	20/01/2014 16:54	C Source File	4 KB
Carica_partita.o	20/01/2014 16:54	File O	379 KB
costanti	23/01/2014 12:25	C Header File	1 KB
funzioni	24/01/2014 02:02	C Source File	2 KB
funzioni.o	24/01/2014 02:02	File O	374 KB
Impostazioni	20/01/2014 16:50	C Source File	2 KB
Impostazioni.o	20/01/2014 16:50	File O	375 KB
libreria	22/01/2014 10:29	C Header File	2 KB
main	20/01/2014 15:41	C Source File	2 KB
main.o	20/01/2014 16:47	File O	374 KB
Makefile.win	25/01/2014 22:16	File WIN	2 KB
Nuova_partita	23/01/2014 12:25	C Source File	21 KB
Nuova_partita.o	23/01/2014 12:26	File O	399 KB
Partite_salvate	25/01/2014 22:20	File BIN	0 KB
Record	22/01/2014 10:30	C Source File	4 KB
Record.o	22/01/2014 10:30	File O	378 KB
Record_file	25/01/2014 22:20	File BIN	1 KB
Salva_partita	24/01/2014 02:06	C Source File	7 KB
Salva_partita.o	24/01/2014 02:06	File O	382 KB
Scale_e_serpenti	05/12/2013 20:57	Dev-C++ Project ...	2 KB
Scale_e_serpenti	25/01/2014 22:16	Applicazione	2.667 KB
Scale_e_serpenti.layout	25/01/2014 22:16	File LAYOUT	1 KB

\*\*\*Record\_file ha comunque dimensione di 1 kb ma solo perchè ha il numero di giocatore nella colonna del giocatore , ma non esiste alcuna partita vinta o fatta associata a nessun giocatore.\*\*\*

## Nuova\_partita()

### Dati di input

Nome	Descrizione	Vincoli
cella_g[GIOCATORI_MAX]	indica la cella del giocatore	intero da [0-100]
vincitore	indica il vincitore	intero da [1-4]
num	indica il turno del giocatore	intero da [0-3]
n_partecipanti	indica il numero di partecipanti	intero da [2-4]
Partite_salvate	file sul quale ci sono le partite salvate	Binario [pieno,vuoto]
Record_file	file sul quale ci sono i record dei giocatori	Binario [pieno,vuoto]

GIOCATORI\_MAX 1 classe

Record\_file 2 classi di equivalenza (pieno,vuoto)

Partite\_salvate 2 casi di equivalenza (pieno,vuoto)

num 4 classi di equivalenza (0,1,2,3)

vincitore 4 classi di equivalenza (0,1,2,3,4)

cella\_g[GIOCATORI\_MAX] 100 classi di equivalenza (0...100)

n\_partecipanti 3 classi di equivalenza (2,3,4)

totale dei casi :  $100 \cdot 4 \cdot 4 \cdot 2 \cdot 2 \cdot 3 = 19200$

ridotti a 4

### Casi di test:

#### Caso 1:

GIOCATORI\_MAX= 4

Record\_file = vuoto

Partite\_salvate =pieno

num =1

n\_partecipanti=2

cella\_g[num ]=100

vincitore=1

output " il giocatore 1 ha vinto, viene aggiornato il file Record\_file"

100 ♥	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25 ♦	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3

**LEGENDA**

**TASTI**

Esc    Torna al menu'

Invio    Lancio dado

**SIMBOLI**

♥    Giocatore 1

♦    Giocatore 2

▲    Giocatore 3

▼    Giocatore 4

▼    Serpente

▲    Scala

**PARTECIPANTI**

♥    ♦

Turno del Giocatore 1

Fai la scelta:

Lancio dado...

Valore del dado: 5

**GIOCATORE 1 HA VINTO!\_**

Record\_file

RECORD		
GIOCATORE #	PARTITE VINTE	PARTITE GIOCATE
1	6	18
4	5	5
3	4	9
2	3	18

Premi ESC per tornare al menu' :

### Caso 2:

GIOCATORI\_MAX= 4

Record\_file = pieno

Partite\_salvate =pieno

num =1

n\_partecipanti=2

cella\_g[num ]=10

vincitore=0

output " il giocatore 1 sulla scala"

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13◆	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3♥

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI

- ♥ Giocatore 1
- ◆ Giocatore 2
- ♣ Giocatore 3
- ♠ Giocatore 4
- ▼ Serpente
- ▲ Scala

PARTECIPANTI

♥ ◆  
 Turno del Giocatore 1  
 Fai la scelta:  
 Lancio dado...  
 Valore del dado: 2

GIOCATORE 1 SULLA SCALA!

### Caso 3:

GIOCATORI\_MAX= 4

```
Record_file = pieno
Partite_salvate = pieno
num = 2
n_partecipanti = 2
cella_g[num] = 46
vincitore = 0
output " il giocatore 2 sul serpente"
```

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3

LEGENDA

TASTI

Esc

Torna al menu'

Invio

Lancio dado

SIMBOLI

♥

Giocatore 1

♦

Giocatore 2

♣

Giocatore 3

♠

Giocatore 4

▼

Serpente

▲

Scala

PARTECIPANTI

♥♦

Turno del Giocatore 2

Fai la scelta:

Lancio dado...

Valore del dado: 4

GIOCATORE 2 SUL SERPENTE\_

#### Caso 4:

```
GIOCATORI_MAX = 4
Record_file = pieno
Partite_salvate = pieno
num = 1
n_partecipanti = 2
cella_g[num] = 9
vincitore = 0
output " turno del giocatore 1"
```

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8 ◆	9 ♥	10▲3

LEGENDA

TASTI

Esc

Torna al menu'

Invio

Lancio dado

SIMBOLI

♥

Giocatore 1

◆

Giocatore 2

♣

Giocatore 3

♠

Giocatore 4

▼

Serpente

▲

Scala

PARTECIPANTI

♥ ◆

Turno del Giocatore 1

Fai la scelta:

## Grafica\_Nuova\_partita()

Dati di input

Nome

Descrizione

Vincoli

n\_partecipanti

quantità di partecipanti alla partita intero [2-4]

DIM\_CAMPO

Dimensione del campo da gioco

n\_partecipanti 3 classi di equivalenza [2] [3] [4]

DIM\_CAMPO 1 classe di equivalenza [100]

totale casi:  $3 \times 1 = 3$

Casi di test:

Caso 1:

n\_partecipanti= 2

DIM\_CAMPO =100

output "stampa a video il campo da gioco con 2 giocatori"

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3

LEGENDA

TASTI

Esc Torna al menu'

Invio Lancio dado

SIMBOLI

♥ Giocatore 1

♦ Giocatore 2

♣ Giocatore 3

♠ Giocatore 4

▼ Serpente

▲ Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 1

Fai la scelta:

♥ ♦

## Caso 2:

n\_partecipanti= 3

DIM\_CAMPO =100

output "stampa a video il campo da gioco con 3 giocatori"



100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI

Esc

Torna al menu'

Invio

Lancio dado

SIMBOLI

♥

Giocatore 1

♦

Giocatore 2

♣

Giocatore 3

♠

Giocatore 4

▼

Serpente

▲

Scala

PARTECIPANTI

♥ ♦ ♣ ♠

Turno del Giocatore 1

Fai la scelta: \_

♥♦♣♠

### Caso 3:

n\_partecipanti= 4

DIM\_CAMPO =100

output "stampa a video il campo da gioco con 4 giocatori"

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI

Esc

Torna al menu'

Invio

Lancio dado

SIMBOLI

♥

Giocatore 1

♦

Giocatore 2

♣

Giocatore 3

♠

Giocatore 4

▼

Serpente

▲

Scala

PARTECIPANTI

♥ ♦ ♣ ♠

Turno del Giocatore 1

Fai la scelta:

♥♦♣♠

## Inizializzare\_partita()

### Dati di input

Nome	Descrizione	Vincoli
cella_g[GIOCATORI_MAX]	indica la cella del giocatore	intero da [0-100]
n_partecipanti	quantità di partecipanti alla partita	intero [2-4]
num	indica il turno del giocatore	intero da [0-3]
GIOCATORI_MAX	costante globale che indica la quantità di giocatori massimi	4

cella\_g[GIOCATORI\_MAX] 100 classi di equivalenza (0....100)

GIOCATORI\_MAX 1 classe

num 4 classi di equivalenza (0,1,2,3)

n\_partecipanti 3 classi di equivalenza (2,3,4)

totale casi di test:  $100 * 1 * 4 * 3 = 1200$

### Casi di test:

#### Caso 1:

cella\_g[num]=20

GIOCATORI\_MAX=4

num=3

n\_partecipanti=3

output "setta a 0 tutte le posizioni per quanti sono i partecipanti e setta a 0 anche num e vincitore e chiama Grafica\_Nuova\_partita per stampare il campo da gioco"

#### Caso 2:

cella\_g[num]=39

GIOCATORI\_MAX=4

num=2

n\_partecipanti=2

output "setta a 0 tutte le posizioni per quanti sono i partecipanti e setta a 0 anche num e vincitore e chiama Grafica\_Nuova\_partita per stampare il campo da gioco"

#### Caso 3:

cella\_g[num]=20

GIOCATORI\_MAX=4

num=0

n\_partecipanti=2

output "setta a 0 tutte le posizioni per quanti sono i partecipanti e setta a 0 anche num e vincitore e chiama Grafica\_Nuova\_partita per stampare il campo da gioco"

ecc..

\*\*\*Non ci sono schermate in quanto è una modifica dal punto di vista non grafico, infatti in Grafica\_Nuova\_partita ci sono le schermate attinenti anche ad inizializza\_partita\*\*\*

## Spostare\_pedina()

### Dati di input

Nome	Descrizione	Vincoli
scelta_tasto	tasto scelto dall'utente	intero [INVIO,ESC]
dado	variabile locale che contiene il valore casuale generato	intero[1,MAXDADO]
MAXDADO	variabile che contiene il max della generazione	intero positivo
cella_g[GIOCATORI_MAX]	indica la cella del giocatore	intero da [0-100]
num	indica il turno del giocatore	intero da [0-3]
n_partecipanti	quantità di partecipanti alla partita	intero [2-4]
vincitore	indica il vincitore	intero da [1-4]
GIOCATORI_MAX	costante globale che indica la quantità di giocatori massimi	4

scelta\_tasto    255 classi di equivalenza    tutti i tasti della tastiera  
dado            2 classi di equivalenza            (1,MAXDADO)  
MAXDADO       1 classe di equivalenza  
cella\_g[GIOCATORI\_MAX]    100 classi di equivalenza    (0....100)  
num    4 classi di equivalenza    (0,1,2,3)  
n\_partecipanti    3 classi di equivalenza    (2,3,4)  
vincitore    4 classi di equivalenza    (0,1,2,3,4)  
GIOCATORI\_MAX    1 classe  
totale casi :  $255*2*1*100*4*3*4*1=2.448.000$

### Casi di test

#### Caso 1:

scelta\_tasto=INVIO  
dado=5  
MAXDADO=6  
cella\_g[num]=5  
num=2  
n\_partecipanti=2  
vincitore=0  
GIOCATORI\_MAX=4  
output "genera il valore del dado associato al giocatore 2, incrementa del valore uscito al dado  
cella\_g[num] e inoltre chiama la funzione Grafica\_cancellare\_pedina,Grafica\_muovere\_pedina,  
Verificare\_vittoria, Vedere serpente,Vedere\_scala"

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4♥	5♦	6	7▲3	8	9	10▲3

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
▲	Giocatore 3
▼	Giocatore 4
▼	Serpente
▲	Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 2

Fai la scelta:

Lancio dado...

Valore del dado: 5

### Caso 2:

scelta\_tasto=ESC

dado=5

MAXDADO=6

cella\_g[num]=5

num=2

n\_partecipanti=2

vincitore=0

GIOCATORI\_MAX=4

output "torna al menù principale"

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4♥	5♦	6	7 3	8	9	10▲3

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 1

Fai la scelta:

<pre> ----- ***  ***  **  *  ***  ***  ***  ***  ***  ***  ***  *** *   *   *  *  *  *   *   *   *   *   *   *   *   *   * *  *   *  *  *  *  *   *   *   *   *   *   *   *   * *  *  *  *  *  *  *   *   *   *   *   *   *   *   * ****  **** *   *  **** ****  ****  **** *   *   *   * ----- </pre>
<pre> [1]. Nuova partita; [2]. Carica partita; [3]. Salva partita; [4]. Impostazioni; [5]. Record;  [0]. Fine.  Fai la tua scelta premendo un tasto fra quelli indicati nel menu': _ </pre>

```

Caso 3:
scelta_tasto= f
dado=5(quello precedente)
MAXDADO=6
cella_g[num]=30
num=2
n_partecipanti=2
vincitore=0
GIOCATORI_MAX=4
output "tasto non valido e quindi fa inserire un altro tasto che sia o invio o ESC"

```

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3

LEGENDA

TASTI

Esc

Torna al menu'

Invio

Lancio dado

SIMBOLI

♥

Giocatore 1

♦

Giocatore 2

♣

Giocatore 3

♠

Giocatore 4

▼

Serpente

▲

Scala

PARTECIPANTI

♥♦

Turno del Giocatore 1

Fai la scelta: \_

♥♦

#### Caso 4:

scelto\_tasto=INVIO

dado=5

MAXDADO=6

cella\_g[num]=100

num=2

n\_partecipanti=2

vincitore=2

GIOCATORI\_MAX=4

output "genera il valore del dado associato al giocatore 2 incrementa di 6 cella\_g[num] e chiama Grafica\_cancellare\_pedina e Grafica\_muovere pedina e inoltre scrive che il giocatore 2 ha vinto"

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3

**LEGENDA**

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

**PARTECIPANTI**

♥ ♦  
 Turno del Giocatore 2  
 Fai la scelta:  
 Lancio dado...  
 Valore del dado: 5

**GIOCATORE 2 HA VINTO!**

#### Caso 5:

scelto\_tasto=INVIO

dado=2

MAXDADO=6

cella\_g[num]=10

num=1

n\_partecipanti=2

vincitore=0

GIOCATORI\_MAX=4

output "giocatore 1 sulla scala viene chiamata Grafica\_cancellare\_pedina e Grafica\_muovere\_pedina per spostare la pedina cella\_g[num] diventa 76"

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 1

Fai la scelta:

Lancio dado...

Valore del dado: 6

GIOCATORE 1 SULLA SCALA!

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 2

Fai la scelta: \_

#### Caso 6:

scelto\_tasto=INVIO

dado=6

MAXDADO=6

cella\_g[num]=88

num=2

n\_partecipanti=2



vincitore=0

GIOCATORI\_MAX=4

output "giocatore 2 sul serpente chiamata Grafica\_cancellare\_pedina e Grafica\_muovere\_pedina per spostare la pedina cella\_g[num] diventa 23 "

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI

Esc Torna al menu'

Invio Lancio dado

SIMBOLI

♥ Giocatore 1

♦ Giocatore 2

♣ Giocatore 3

♠ Giocatore 4

▼ Serpente

▲ Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 2

Fai la scelta:

Lancio dado...

Valore del dado: 3

GIOCATORE 2 SUL SERPENTE\_

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI

Esc Torna al menu'

Invio Lancio dado

SIMBOLI

♥ Giocatore 1

♦ Giocatore 2

♣ Giocatore 3

♠ Giocatore 4

▼ Serpente

▲ Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 1

Fai la scelta:

Caso 7:

scelto\_tasto= f

dado=5

MAXDADO=6

```

cella_g[num]=96
num=2
n_partecipanti=2
vincitore=0
GIOCATORI_MAX=4
output "non è valido il valore di scelta_tasto e quindi resta la stessa schermata in attesa di un nuovo
input"

```

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	835	84	85	86	87	88▼2	89	90
80	79	78	77▼3	763	75	74	73	72	71
61	62	63	641	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	521	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	382	37	364	35	34▲4	33	32	31
21	22	232	24	25	26	27	28	295	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	73	8	9	10▲3

**LEGENDA**

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

**PARTECIPANTI**

♥ ♦  
Turno del Giocatore 1  
Fai la scelta:

## Salvare\_risultati()

### Dati di input

Nome	Descrizione	Vincoli
vincitore	indica il vincitore	intero da [1-4]
Record_file	file sul quale ci sono i record dei giocatori	Binario [pieno,vuoto]
n_partecipanti	quantità di partecipanti alla partita	intero [2-4]

vincitore 3 classi di equivalenza [1][2][3][4]  
Record\_file 2 classi di equivalenza [pieno,vuoto]  
n\_partecipanti 3 classi di equivalenza [0][1][2][3]  
totale casi di test:  $3 \cdot 3 \cdot 2 = 18$

### CASI DI TEST

#### Caso 1:

```

vincitore=1
Record_file=vuoto
n_partecipanti=2
output "scrive i risultati all'inizio del file con il numero di vincitore,incrementa la quantità di partite vinte
del vincitore e quelle fatte per tutti i partecipanti"

```

### Caso 2:

vincitore=2

Record\_file=pieno

n\_partecipanti=3

output "scrive i risultati alla fine del file con il numero di vincitore, incrementa la quantità di partite vinte del vincitore e quelle fatte per tutti i partecipanti"

### Caso 3:

vincitore=3

Record\_file=vuoto

n\_partecipanti=4

output "scrive i risultati all'inizio del file con il numero di vincitore, incrementa la quantità di partite vinte del vincitore e quelle fatte per tutti i partecipanti"

### caso 4:

vincitore=4

Record\_file=vuoto

n\_partecipanti=4

output "scrive i risultati all'inizio del file con il numero di vincitore, incrementa la quantità di partite vinte del vincitore e quelle fatte per tutti i partecipanti"

**\*\*\*NON E' DIMOSTRABILE GRAFICAMENTE IN QUANTO SONO CAMBIAMENTI INTERNI A Record\_file\*\*\***

## Turno\_giocatore()

### Dati di input

#### Nome

s

num

#### Descrizione

variabile locale che indica il carattere letto

indica il turno del giocatore globale

#### Vincoli

intero [INVIO,ESC]

intero da [0-3]

s 255 classi di equivalenza tutti i tasti della tastiera

num 4 classi di equivalenza (0,1,2,3)

totale casi di test:  $4 \cdot 255 = 1020$

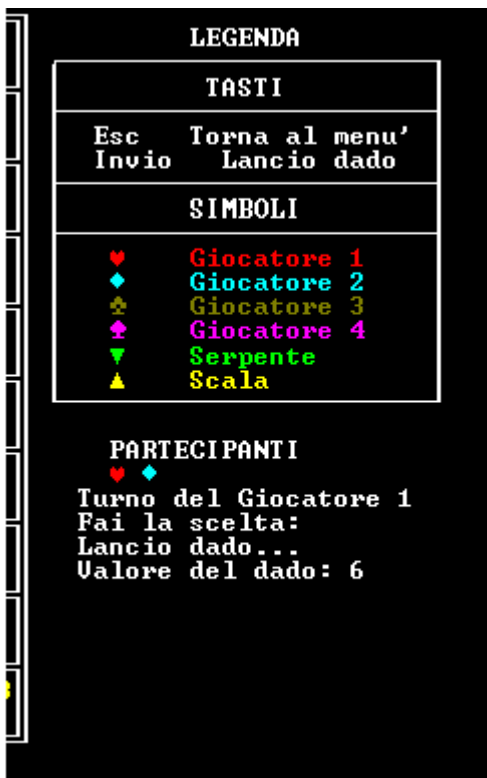
### Casi di test:

#### Caso 1

s= INVIO

num=0

output "scrittura lancio dado .. e restituisce il valore inserito "



## Caso 2

s=ESC

num=1

output "torna al menù iniziale "



## Caso 3

s= (

num=2

output "non accetta il valore e resta la stessa schermata per far inserire un altro valore di s"

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86◆	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58♥	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

#### LEGENDA

##### TASTI

Esc Torna al menu'  
Invio Lancio dado

##### SIMBOLI

♥ Giocatore 1  
◆ Giocatore 2  
♣ Giocatore 3  
♠ Giocatore 4  
▼ Serpente  
▲ Scala

##### PARTECIPANTI

♥ ◆  
Turno del Giocatore 2  
Fai la scelta:

#### Caso 4

s= 2

num=3

output "non accetta il valore e resta la stessa schermata per far inserire un altro valore di s "

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86◆	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58♥	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

#### LEGENDA

##### TASTI

Esc Torna al menu'  
Invio Lancio dado

##### SIMBOLI

♥ Giocatore 1  
◆ Giocatore 2  
♣ Giocatore 3  
♠ Giocatore 4  
▼ Serpente  
▲ Scala

##### PARTECIPANTI

♥ ◆  
Turno del Giocatore 2  
Fai la scelta:

#### Caso 5

s= 7

num=2

output "non accetta il valore e resta la stessa schermata per far inserire un altro valore di s"

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86◆	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58♥	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3

**LEGENDA**

**TASTI**

Esc    Torna al menu'

Invio    Lancio dado

**SIMBOLI**

♥    Giocatore 1

◆    Giocatore 2

♣    Giocatore 3

♠    Giocatore 4

▼    Serpente

▲    Scala

**PARTECIPANTI**

♥    ◆

Turno del Giocatore 2

Fai la scelta:

ecc..

## Vedere\_scala()

### Dati di input

Nome	Descrizione	Vincoli
cella_g[GIOCATORI_MAX]	indica la cella del giocatore	intero da [0-100]
GIOCATORI_MAX	costante globale che indica la quantità di giocatori massimi	4
num	Variabile locale che è una stringa	stringa di caratteri con max 4 caratteri
ELEM_V	Quantità elementi del vettore scala	intero 5
v_scala_partenza(20, 17, 10, 34, 59)	Sequenza di interi di 5 elementi	
v_scala_arrivo(64, 38, 76, 95, 83)	Sequenza di interi di 5 elementi	
v_scala_partenza(20, 17, 10, 34, 59)	5 classi di equivalenza	
v_scala_arrivo(64, 38, 76, 95, 83)	5 classi di equivalenza	
cella_g[GIOCATORI_MAX]	100 classi di equivalenza (0....100)	
GIOCATORI_MAX	1 classe	
num	4 classi di equivalenza (0,1,2,3)	
ELEM_V	1 classe di equivalenza	
totale dei casi = 100*1*4*1*10*10=40.000		

### Caso 1:

v\_scala\_partenza(20, 17, 10, 34, 59)

v\_scala\_arrivo(64, 38, 76, 95, 83)

cella\_g[num]=9

num=2

ELEM\_V = 5

GIOCATORI\_MAX=4

output "non è su una scala e quindi il gioco prosegue "

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI

Esc    Torna al menu'

Invio    Lancio dado

SIMBOLI

♥    Giocatore 1

♦    Giocatore 2

♣    Giocatore 3

♠    Giocatore 4

▼    Serpente

▲    Scala

PARTECIPANTI

♥    ♦

Turno del Giocatore 2

Fai la scelta:

## Caso 2:

v\_scala\_partenza(20, 17, 10, 34, 59)

v\_scala\_arrivo(64, 38, 76, 95, 83)

cella\_g[num]=59

num=1

ELEM\_V = 5

GIOCATORI\_MAX=4

output "il giocatore 1 è sulla scala e quindi la variabile di lavoro rilevato viene settata uguale a vero e viene spostata la pedina graficamente e cancellata quella in posizione precedente richiamando le funzioni apposite e passando la variabile di lavoro cella\_prec\_g[num] che contiene la posizione vecchia della pedina"

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68 ◆	69	70
60	59▲5 ♥	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI

- ♥ Giocatore 1
- ◆ Giocatore 2
- ♣ Giocatore 3
- ♠ Giocatore 4
- ▼ Serpente
- ▲ Scala

PARTECIPANTI

♥ ◆  
 Turno del Giocatore 1  
 Fai la scelta:  
 Lancio dado...  
 Valore del dado: 5

GIOCATORE 1 SULLA SCALA!

ecc..



## Vedere\_serpente()

### Dati di input

Nome	Descrizione	Vincoli
cella_g[GIOCATORI_MAX]	indica la cella del giocatore	intero da [0-100]
num	indica il turno del giocatore	intero da [0-3]
ELEM_V	quantità elementi nei vettori dei serpenti	
GIOCATORI_MAX	costante globale che indica la quantità di giocatori massimi	4
V_serpente_partenza=(96, 88, 77, 46, 67)	Sequenza di interi di 5 elementi	
V_serpente_arrivo=(52, 23, 7, 36, 29)	Sequenza di interi di 5 elementi	

V\_serpente\_partenza=(96, 88, 77, 46, 67) 5 classi di equivalenza

V\_serpente\_arrivo=(52, 23, 7, 36, 29) 5 classi di equivalenza

cella\_g[GIOCATORI\_MAX] 100 classi di equivalenza (0....100)

GIOCATORI\_MAX 1 classe

num 4 classi di equivalenza (0,1,2,3)

ELEM\_V 1 classe di equivalenza

totale dei casi =  $100 * 1 * 4 * 1 * 10 * 10 = 40.000$

### Caso 1:

V\_serpente\_partenza=(96, 88, 77, 46, 67)

V\_serpente\_arrivo=(52, 23, 7, 36, 29)

cella\_g[num]=9

num=2

ELEM\_V = 5

GIOCATORI\_MAX=4

output "non è su un serpente e quindi il gioco prosegue "

100	99	98	97	96▼1	954	94	93	92	91
81	82	835	84	85	86	87	88▼2	89	90
80	79	78	77▼3	763	75	74	73	72	71
61	62	63	641	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	521	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	382	37	364	35	34▲4	33	32	31
21	22	232	24	25	26	27	28	295	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5▼	6	73	8	9▼	10▲3

**LEGENDA**

**TASTI**

Esc	Torna al menu'
Invio	Lancio dado

**SIMBOLI**

♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

**PARTECIPANTI**

♥♦	Turno del Giocatore 2
	Fai la scelta:

### Caso 2:

V\_serpente\_partenza=(96, 88, 77, 46, 67)

V\_serpente\_arrivo=(52, 23, 7, 36, 29)

cella\_g[num]=46

num=1

ELEM\_V = 5

GIOCATORI\_MAX=4

output "il giocatore 1 è sul serpente e quindi la variabile di lavoro rilevato viene settata uguale a vero e viene spostata la pedina graficamente e cancellata quella in posizione precedente richiamando le funzioni apposite e passando la variabile di lavoro cella\_prec\_g[num] che contiene la posizione vecchia della pedina"

100	99	98	97	96▼1	954	94	93	92	91
81	82	835	84	85	86	87	88▼2	89	90
80	79	78	77▼3	763	75	74	73	72	71
61	62	63	641	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55◆	54	53	521	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	382	37	364	35	34▲4	33	32	31
21	22	232	24	25	26	27	28	295	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	73	8	9	10▲3

LEGENDA

TASTI

EscTorna al menu'  
InvioLancio dado

SIMBOLI

♥Giocatore 1  
◆Giocatore 2  
♣Giocatore 3  
♠Giocatore 4  
▼Serpente  
▲Scala

PARTECIPANTI

♥◆  
Turno del Giocatore 1  
Fai la scelta:  
Lancio dado...  
Valore del dado: 1

GIOCATORE 1 SUL SERPENTE\_

ecc..

## Vedere\_vittoria()

### Dati di input

#### Nome

cella\_g[GIOCATORI\_MAX]

num

DIM\_CAMPO

GIOCATORI\_MAX

#### Descrizione

indica la cella del giocatore

indica il turno del giocatore

dimensione del campo da gioco

costante globale che indica la quantità di giocatori massimi 4

#### Vincoli

intero da [0-100]

intero da [0-3]

intero 100

DIM\_CAMPO 1 classe di equivalenza

cella\_g[GIOCATORI\_MAX] 100 classi di equivalenza (0....100)

num 4 classi di equivalenza (0,1,2,3)

GIOCATORI\_MAX 1 classe di equivalenza

totale casi : 4\*100\*1\*1=400

### Casi di test

#### Caso 1:

DIM\_CAMPO=100

cella\_g[num]=76

num=2

GIOCATORI\_MAX=4

output "non c'è alcun vincitore il gioco prosegue"

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI

♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 2

Fai la scelta: \_

### Caso 2:

DIM\_CAMPO=100

cella\_g[num]=100

num=2

GIOCATORI\_MAX=4

output "il giocatore 2 ha vinto , viene assegnato alla variabile vincitore il giocatore che ha vinto incrementato di 1"

100	99	98	97	96▼1	954	94	93	92	91
81	82	835	84	85	86	87	88▼2	89	90
80	79	78	77▼3	763	75	74	73	72	71
61	62	63	641	65	66	67▼5	68	69	70
60	59▲5	58	57	56♥	55	54	53	521	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	382	37	364	35	34▲4	33	32	31
21	22	232	24	25	26	27	28	295	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	73	8	9	10▲3

LEGENDA

TASTI

Esc Torna al menu'

Invio Lancio dado

SIMBOLI

♥ Giocatore 1

♦ Giocatore 2

♣ Giocatore 3

♠ Giocatore 4

▼ Serpente

▲ Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 2

Fai la scelta:

Lancio dado...

Valore del dado: 5

GIOCATORE 2 HA VINTO!

## Grafica\_cancellare\_pedina()

### Dati di input

Nome	Descrizione	Vincoli
cella_g[GIOCATORI_MAX]	indica la cella del giocatore	intero da [0-100]
cella_g_prec[GIOCATORI_MAX]	vettore che contiene tutte le posiz.prec di ogni giocatore	intero da [0-100]
num	indica il turno del giocatore	intero da [0-3]
DIM_CAMPO	dimensione del campo da gioco	intero 100

num 4 classi di equivalenza (0,1,2,3)

DIM\_CAMPO 1 classe di equivalenza

cella\_g[GIOCATORI\_MAX] 100 classi di equivalenza (0....100)

cella\_prec\_g[GIOCATORI\_MAX] 100 classi di equivalenza (0....100)

totale casi :  $100 \cdot 100 \cdot 4 \cdot 1 = 40.000$

### Casi di test:

#### caso 1:

num=2

DIM\_CAMPO=100

cella\_g[num]=8

cella\_prec\_g[num]=7

output "cancella la pedina nella che sta nella cella\_prec\_g[num] cioè la 7 in questo caso"

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 2  
Fai la scelta:

caso 2:

num=1

DIM\_CAMPO=100

cella\_g[num]=76

cella\_prec\_g[num]=10

output "cancella la pedina nella che sta nella cella\_prec\_g[num] cioè la 10 in questo caso"

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	♦	44	45	46▼4	47	48	49
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

LEGENDA

TASTI	
Esc	Torna al menu'
Invio	Lancio dado

SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

PARTECIPANTI

♥ ♦

Turno del Giocatore 1  
Fai la scelta:  
Lancio dado...  
Valore del dado: 2

GIOCATORE 1 SULLA SCALA!

ECC..

## Grafica\_muovere\_pedina()

### Dati di input

Nome	Descrizione
cella_g[GIOCATORI_MAX]	indica la cella del giocatore
num	indica il turno del giocatore
DIM_CAMPO	dimensione del campo da gioco

### Vincoli

intero da [0-100]  
intero da [0-3]  
intero 100

GIOCATORI\_MAX 1 classe di equivalenza

cella\_g[GIOCATORI\_MAX] 100 classi di equivalenza (0....100)

DIM\_CAMPO 1 classe di equivalenza

totale casi :  $1 \cdot 100 \cdot 1 = 100$

### Casi di test:

#### caso 1:

num=2

DIM\_CAMPO=100

cella\_g[num]=4

output "scrive la pedina nella che sta nella cella\_g[num]cioè nella 4 in questo caso"

100	99	98	97	96▼1	95 4	94	93	92	91
81	82	83 5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76 3	75	74	73	72	71
61	62	63	64 1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52 1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38 2	37	36 4	35	34▲4	33	32	31
21	22	23 2	24	25	26	27	28	29 5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7 3	8	9	10▲3

**LEGENDA**

TASTI	
Esc	Torna al menu'
Invio	Lancio dado
SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

**PARTECIPANTI**

♥ ♦  
Turno del Giocatore 1  
Fai la scelta:

#### caso 2:

num=2

DIM\_CAMPO=100

cella\_g[num]=7

output "scrive la pedina nella che sta nella cella\_g[num]cioè nella 7 in questo caso"  
ecc..

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	7▲3	8	9	10▲3

LEGENDA

TASTI

Esc

Torna al menu'

Invio

Lancio dado

SIMBOLI

♥

Giocatore 1

♦

Giocatore 2

♣

Giocatore 3

♠

Giocatore 4

▼

Serpente

▲

Scala

PARTECIPANTI

♥♦

Turno del Giocatore 1

Fai la scelta:

## Carica\_partita()

### Dati di input

Nome	Descrizione	Vincoli
nome_scelto	nome inserito var locale	carattere [5]
size	dimensione del file "Partite_salvate"	long > 0
scelta_carica	scelta dell'utente se tornare indietro o no	intero[ s,ESC]

scelta\_carica 255 classi di equivalenza

size 1 classe di equivalenza [0-+00]

nome\_scelto 1 classe di equivalenza

totale casi = 255

### CASI DI TEST

#### caso 1:

scelta\_carica=s

size=0

nome\_scelto=frank

output " nome partita non esistente"

CARICA PARTITA					
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="text-align: center; padding: 5px;">NOME PARTITA</th> <th style="text-align: center; padding: 5px;"># PARTECIPANTI</th> </tr> <tr> <td style="text-align: center; padding: 5px;">ciao</td> <td style="text-align: center; padding: 5px;">2</td> </tr> </table>	NOME PARTITA	# PARTECIPANTI	ciao	2	<p>Inserire il nome della partita da caricare e premere invio: frank (MAX 5 LETTERE!)</p> <p style="color: red; text-align: center;">NOME PARTITA NON ESISTENTE!_</p>
NOME PARTITA	# PARTECIPANTI				
ciao	2				

**caso 2:**

scelta\_carica=ESC

size=0

nome\_scelto=""

output "torna al menu principale"

CARICA PARTITA
<p>Uuoi caricare una partita salvata?</p> <p style="text-align: center;">s = SI                      ESC = Indietro</p> <p>Fai la tua scelta:</p>

<pre> ***  ***  **  *  ***  ***  ***  ***  ****  *****  *  *  *****  * * *  * * ***  ***  *  *  ***  ***  ***  ***  ***  ***  *  *  *  *  *  *  * </pre>
<pre> [1]. Nuova partita; [2]. Carica partita; [3]. Salva partita; [4]. Impostazioni; [5]. Record;  [0]. Fine. </pre> <p>Fai la tua scelta premendo un tasto fra quelli indicati nel menu':</p>

**caso 3:**

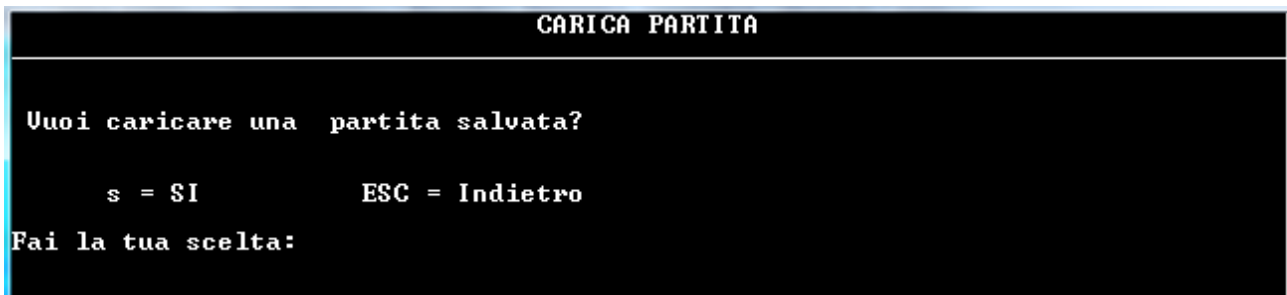
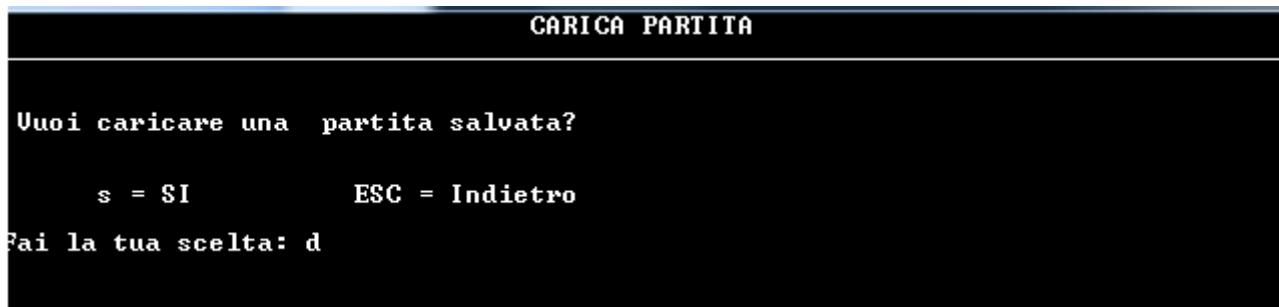
scelta\_carica=d

size=0

nome\_scelto=""

output "non accetta il carattere inserito nel scelta\_carica"





caso 4 :

scelta\_carica=s

size=280

nome\_scelto=frank

output "legge i dati dal file Partite\_salvate e riprende la partita"





PARTECIPANTI  
 Turno del Giocatore 1  
 Fai la scelta:

```
scelta_carica=s
size=0
nome_scelto=frank
output "attenzione, non ci sono partite da caricare"
```

ecc..

### Dati di input

Nome	Descrizione	Vincoli
num	indica il turno del giocatore	intero da [0-3]
n_partecipanti	quantità di partecipanti alla partita	intero [2-4]

num 4 classi di equivalenza (0,1,2,3)  
n\_partecipanti 3 classi di equivalenza (2,3,4)

totale casi :  $4 \times 3 = 12$

#### casi di test:

##### caso 1:

num=1

n\_partecipanti=2

output "scrive le pedine in base alla quantità di pedine della partita ricaricata nelle loro posizioni chiamando grafica\_muovere\_pedina"

100	99	98	97	96▼1	954	94	93	92	91
81	82	835	84	85	86	87	88▼2	89	90
80	79	78	77▼3	763	75	74	73	72	71
61	62	63	641	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	521	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	382	37	364	35	34▲4	33	32	31
21	22	232	24	25	26	27	28	295	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2	3	4	5	6	73	8	9	10▲3

LEGENDA	
TASTI	
Esc	Torna al menu'
Invio	Lancio dado
SIMBOLI	
♥	Giocatore 1
♦	Giocatore 2
♣	Giocatore 3
♠	Giocatore 4
▼	Serpente
▲	Scala

#### PARTECIPANTI

♥ ♦  
Turno del Giocatore 1  
Fai la scelta: \_

##### caso 2:

num=1

n\_partecipanti=3

output "scrive le pedine in base alla quantità di pedine della partita ricaricata nelle loro posizioni  
grafica\_muovere\_pedina "

100	99	98	97	96▼1	95▲4	94	93	92	91
81	82	83▲5	84	85	86	87	88▼2	89	90
80	79	78	77▼3	76▲3	75	74	73	72	71
61	62	63	64▲1	65	66	67▼5	68	69	70
60	59▲5	58	57	56	55	54	53	52▲1	51
41	42	43	44	45	46▼4	47	48	49	50
40	39	38▲2	37	36▲4	35	34▲4	33	32	31
21	22	23▲2	24	25	26	27	28	29▲5	30
20▲1	19	18	17▲2	16	15	14	13	12	11
1	2▲	3	4	5♥♣	6	7▲3	8	9	10▲3

**LEGENDA**

**TASTI**

Esc      Torna al menu'

Invio    Lancio dado

**SIMBOLI**

♥      Giocatore 1

♦      Giocatore 2

♣      Giocatore 3

♠      Giocatore 4

▼      Serpente

▲      Scala

**PARTECIPANTI**

♥ ♦ ♣

Turno del Giocatore 1

Fai la scelta: \_

## Salva\_Partita()

### Dati di input

Nome	Descrizione	Vincoli
num	indica il turno del giocatore	intero da [0-3]
n_partecipanti	quantità di partecipanti alla partita	intero [2-4]
nome_scelto	nome inserito var locale	carattere [5]
scelta_carica	scelta dell'utente se tornare indietro o no	intero[ INVIO,ESC]
GIOCATORI_MAX	costante globale che indica la quantità di giocatori massimi	4
cella_g[GIOCATORI_MAX]	indica la cella del giocatore	intero da [0-100]

num 4 classi di equivalenza (0,1,2,3)

n\_partecipanti 3 classi di equivalenza (2,3,4)

nome\_scelto 1 classe di equivalenza

scelta\_carica 255 classi di equivalenza tutti i caratteri della tastiera

GIOCATORI\_MAX 1 classe di equivalenza

cella\_g[GIOCATORI\_MAX] 100 classi di equivalenza

Partite\_salvate 2 casi di equivalenza (pieno,vuoto)

Totale casi :  $4 \cdot 3 \cdot 255 \cdot 1 \cdot 100 \cdot 2 = 612.000$

### Casi di test:

#### caso 1:

num=1

n\_partecipanti=2

nome\_scelto=frank

scelta\_carica=s

Partite\_salvate=pieno

cella\_g[num]=30

GIOCATORI\_MAX=4

output "se il nome scelto coincide con 1 nel file Partite salvate allora da errore nome partita già esistente"

SALVA PARTITA					
<table border="1"><thead><tr><th>NOME PARTITA</th><th># PARTECIPANTI</th></tr></thead><tbody><tr><td>frank</td><td>2</td></tr></tbody></table>	NOME PARTITA	# PARTECIPANTI	frank	2	<p>Inserire il nome della partita da salvare e premere invio: frank (MAX 5 LETTERE!)</p> <p><b>NOME PARTITA GIA' ESISTENTE!_</b></p>
NOME PARTITA	# PARTECIPANTI				
frank	2				

caso 2:

num=4

n\_partecipanti=4

nome\_scelto=gioak

scelta\_carica=s

Partite\_salvate=pieno

cella\_g[num]=30

GIOCATORI\_MAX=4

output "se il nome non coincide con nessun altro nome di partita nel file Partite\_salvate allora scrive sul file  
un nuovo record con il num , n\_partecipanti, nome\_scelto e tutte le posizioni di ogni giocatore"

SALVA PARTITA					
<table border="1"><thead><tr><th>NOME PARTITA</th><th># PARTECIPANTI</th></tr></thead><tbody><tr><td>frank</td><td>2</td></tr></tbody></table>	NOME PARTITA	# PARTECIPANTI	frank	2	<p>Inserire il nome della partita da salvare e premere invio: gioak (MAX 5 LETTERE!)</p> <p><b>PARTITA SALVATA CON SUCCESSO!_</b></p>
NOME PARTITA	# PARTECIPANTI				
frank	2				

caso 3:

num=4

n\_partecipanti=4

nome\_scelto=frank

scelta\_carica=ESC

Partite\_salvate=pieno

cella\_g[num]=30

GIOCATORI\_MAX=4

output "siccome scelta\_carica è ESC torna al menu principale"

CARICA PARTITA	
<p>Vuoi caricare una partita salvata?</p> <p>s = SI                      ESC = Indietro</p> <p>Fai la tua scelta:</p>	



#### caso 4:

num=4

n\_partecipanti=4

nome\_scelto=frank

scelta\_carica=s

Partite\_salvate=vuoto

cella\_g[num]=30

GIOCATORI\_MAX=4

output "siccome il file partite\_salvate è vuoto allora scrive direttamente sul file il record con num,  
n\_partecipanti, nome\_scelto e le posizioni di ogni giocatore"



#### caso 5:

num=4

n\_partecipanti=4

nome\_scelto= ESC

scelta\_carica=s

Partite\_salvate=vuoto

cella\_g[num]=30

GIOCATORI\_MAX=4

output "non accetta un ESC come carattere,quindi bisogna almeno inserire una lettera per salvare"

SALVA PARTITA	
	Inserire il nome della partita da salvare e premere invio: <MAX 5 LETTERE!>
NOME PARTITA	# PARTECIPANTI
frank	2
kkkkk	2

SALVA PARTITA	
	Inserire il nome della partita da salvare e premere invio: f <MAX 5 LETTERE!>
NOME PARTITA	# PARTECIPANTI
frank	2
kkkkk	2

PARTITA SALVATA CON SUCCESSO!\_

caso 6:

num=4

n\_partecipanti=4

nome\_scelto= INVIO

scelta\_carica=s

Partite\_salvate=vuoto

cella\_g[num]=30

GIOCATORI\_MAX=4

output "non accetta un invio diretto come carattere,quindi bisogna almeno inserire una lettera per salvare  
e poi premere invio"

SALVA PARTITA	
	Inserire il nome della partita da salvare e premere invio: <MAX 5 LETTERE!>
NOME PARTITA	# PARTECIPANTI
frank	2
kkkkk	2

SALVA PARTITA	
	Inserire il nome della partita da salvare e premere invio: f <MAX 5 LETTERE!>
NOME PARTITA	# PARTECIPANTI
frank	2
kkkkk	2

PARTITA SALVATA CON SUCCESSO!\_

## Grafica\_elenco\_partite()

### Dati di input

#### Nome

nome

n

#### Descrizione

nome della partita

quantità di partecipanti

#### Vincoli

sequenza di caratteri massimo 5

intero positivo[1-4]

nome 1 classe di equivalenza

n 4 classi di equivalenza (1,2,3,4)

totale casi= 4

#### caso 1:

nome = frank

n= 2

output "scrive una tabella a video in cui sta il nome della partita e la quantità dei partecipanti (il nome e il numero di partecipanti varia ogni volta che viene letto un nuovo record dal file "partite\_salvate")

CARICA PARTITA					
<table><tr><th>NOME PARTITA</th><th># PARTECIPANTI</th></tr><tr><td>frank</td><td>2</td></tr></table>	NOME PARTITA	# PARTECIPANTI	frank	2	Inserire il nome della partita da caricare e premere invio: <MAX 5 LETTERE!>
NOME PARTITA	# PARTECIPANTI				
frank	2				

#### caso 2:

nome = gioak

n= 2

output "scrive una tabella a video in cui sta il nome della partita e la quantità dei partecipanti (il nome e il numero di partecipanti varia ogni volta che viene letto un nuovo record dal file "partite\_salvate")

CARICA PARTITA							
<table><tr><th>NOME PARTITA</th><th># PARTECIPANTI</th></tr><tr><td>frank</td><td>2</td></tr><tr><td>gioak</td><td>2</td></tr></table>	NOME PARTITA	# PARTECIPANTI	frank	2	gioak	2	Inserire il nome della partita da caricare e premere invio: <MAX 5 LETTERE!>
NOME PARTITA	# PARTECIPANTI						
frank	2						
gioak	2						

ecc..



## Leggere\_partite\_salvate()

### Dati di input

Nome	Descrizione	Vincoli
scelta_menu	scelta dell'utente	intero [2-3]
Partite_salvate	file sul quale ci sono le partite salvate	Binario [pieno,vuoto]

scelta\_menu 2 classi di equivalenza  
Partite\_salvate 2 casi di equivalenza (pieno,vuoto)  
totale casi di test = 4

### Casi di test:

#### caso 1:

scelta\_menu =2  
Partite\_salvate=pieno  
output "siccome è un carica partita ( 2) allora verranno lette anche le posizioni dei giocatori oltre al nome della partita e del numero dei partecipanti"

#### caso 2:

scelta\_menu=3  
Partite\_salvate=pieno  
output " siccome è un salva partita(3) allora verrà letto dal file solo il numero dei partecipanti e il nome delle partite"

#### caso 3:

scelta\_menu=2  
Partite\_salvate=vuoto  
output "non viene letto nulla"

#### caso 4

scelta\_menu=3  
Partite\_salvate=vuoto  
output "non viene letto nulla "

**\*\*\*NON E' DIMOSTRABILE GRAFICAMENTE , SI DIMOSTRERA GRAFICAMENTE CON CARICA PARTITA O SALVA PARTITA\*\*\***

## Impostazioni()

### Dati di input

Nome	Descrizione	Vincoli
n_partecipanti	quantità di partecipanti alla partita	intero [2-4]
DUE	costante numerica	intero 50
TRE	costante numerica	intero 51
QUATTRO	costante numerica	intero 53

n\_partecipanti 3 classi di equivalenza (50,51,53)  
DUE 1 classe di equivalenza  
TRE 1 classe di equivalenza

QUATTRO      1 classe di equivalenza  
totale casi : 3

**Casi di test:**

**caso 1:**

n\_partecipanti=50

DUE

TRE

QUATTRO

output "verrà assegnato alla variabile n\_partecipanti il valore di 2"

```
IMPOSTAZIONI

Numero Giocatori [2-4].

Inserire il numero dei giocatori: 2
Oppure premi ESC per tornare al menu'
```

**caso 2:**

n\_partecipanti=51

DUE

TRE

QUATTRO

output "verrà assegnato alla variabile n\_partecipanti il valore di 3"

```
IMPOSTAZIONI

Numero Giocatori [2-4].

Inserire il numero dei giocatori: 3
Oppure premi ESC per tornare al menu'
```

**caso 3:**

n\_partecipanti=52

DUE

TRE

QUATTRO

output "verrà assegnato alla variabile n\_partecipanti il valore di 4"

```
IMPOSTAZIONI

Numero Giocatori [2-4].

Inserire il numero dei giocatori: 4
Oppure premi ESC per tornare al menu'
```

## Record()

### Dati di input

#### Nome

Record\_file

s

#### Descrizione

file sul quale ci sono i record

carattere inserito dall'utente

#### Vincoli

Binario [pieno,vuoto]

Booleano [ESC]

s 255 classi di equivalenza

Record\_file 2 classi di equivalenza

TOTALE CASI : 510

### Casi di test

#### caso 1:

s=40

Record\_file= pieno

output "vengono letti i record e stampati a video ma il valore di s nn è valido e resta la schermata di record"

RECORD		
GIOCATORE #	PARTITE UINTE	PARTITE GIOCATE
1	2	3
2	1	3
3	0	0
4	0	0

Preini ESC per tornare al menu' : \_

#### Caso 2

s=8

Record\_file=pieno

output "vengono letti i record, stampati a video e poi ritorna al menu iniziale perchè è stato premuto ESC"

RECORD		
GIOCATORE #	PARTITE VINTE	PARTITE GIOCATE
1	2	3
2	1	3
3	0	0
4	0	0

Premi ESC per tornare al menu': \_

<pre> ***  ***  **  *  ***  ***  ***  ***  ***  ***  ***  *  *  *  *  * * *  * * ***  ***  *  *  ***  ***  ***  ***  ***  ***  *  *  *  *  *  *  * </pre>		
<pre> [1]. Nuova partita; [2]. Carica partita; [3]. Salva partita; [4]. Impostazioni; [5]. Record;  [0]. Fine. </pre>		
<p>Pai la tua scelta premendo un tasto fra quelli indicati nel menu': _</p>		

ecc..

## Ordinamento\_record()

### Dati di input

Nome	Descrizione	Vincoli
giocatori[GICATORI_MAX]	vettore di interi ordinato in base a partite	
partite_vinte[GIOCATORI_MAX]	vettore di interi ordinato decrescente	
partite_fatte[GIOCATORI_MAX]	vettore di interi ordinato in base a partite vinte	
GIOCATORI_MAX	costante globale che indica la quantità di giocatori massimi	4

GIOCATORI\_MAX 1 classe di equivalenza

partite\_vinte[GIOCATORI\_MAX] 100 classi di equivalenza (0,100)

partite\_fatte[GIOCATORI\_MAX] 100 classi di equivalenza (0,100)

GIOCATORI\_MAX 1 classe di equivalenza

giocatori[GIOCATORI\_MAX] 4 classi di equivalenza (1,2,3,4)

totale casi :  $100 \cdot 100 \cdot 1 \cdot 1 \cdot 4 = 40.000$

### Casi di test:

#### caso 1:

GIOCATORI\_MAX =4

partite\_vinte[i]=4

partite\_fatte[i]=5

giocatori[i]= 1

output "il primo più in cima all'ordinamento è giocatore 1 con 4 partite vinte,vengono ordinati tutti i vettori a lui associati"

RECORD		
GIOCATORE #	PARTITE VINTE	PARTITE GIOCATE
1	4	5
2	1	5
3	0	0
4	0	0

Premi ESC per tornare al menu': \_

#### caso 2:

GIOCATORI\_MAX =4

partite\_vinte[i]=6

partite\_fatte[i]=11

giocatori[i]= 2

output "il primo in cima all'ordinamento è il giocatore 2 con 6 partite vinte,vengono ordinati tutti i vettori a lui associati"

RECORD		
GIOCATORE #	PARTITE VINTE	PARTITE GIOCATE
2	6	11
1	5	11
3	0	0
4	0	0

Premi ESC per tornare al menu': \_

#### caso 3:

GIOCATORI\_MAX =4

partite\_vinte[i]=15

partite\_fatte[i]=17

giocatori[i]= 4

output "il primo in cima all'ordinamento è il giocatore 4 con 15 partite vinte,vengono ordinati tutti i vettori a lui associati"

RECORD		
GIOCATORE #	PARTITE VINTE	PARTITE GIOCATE
4	15	17
3	14	29
1	14	56
2	13	56

Premi ESC per tornare al menu': \_

**caso 4:**

GIOCATORI\_MAX =4

partite\_vinte[i]=3

partite\_fatte[i]=3

giocatori[i]= 3

output "il primo in cima all'ordinamento è il giocatore 3 con 3 partite vinte,vengono ordinati tutti i vettori a lui associati"

RECORD		
GIOCATORE #	PARTITE VINTE	PARTITE GIOCATE
3	3	3
1	0	3
4	0	0
2	0	3

Premi ESC per tornare al menu': \_