POLITECNICO
MILANO 1863

Lezione aggiuntiva 1

Forza 4

Scrivere un programma per simulare una partita di Forza4

Il Main permette di gestire le giocate, richiamare alcune funzioni (descritte nelle slide successive) e infine mostrare il vincitore. Lo schema di gioco è rappresentato da una matrice 6x7. Si deve dare la possibilità a due giocatori di poter effettuare la giocata in maniera alternata.

Alcune variabili da inizializzare nel Main

```
N_RIGHE = 6;
N_COLONNE = 7;

T = zeros(N_RIGHE , N_COLONNE);%matrice iniziale
```

Forza 4 - funzioni

Gruppo 1:

Sviluppare la funzione **GIOCA** che permetta al giocatore di selezionare la colonna dove far cadere la propria pedina : il programma deve segnalare se la scelta è ammissibile oppure se la colonna scelta è piena

```
function A_mod = gioca(A , idGiocatore)
```

Gruppo 2:

Sviluppare la funzione **CONTROLLASEVINCE** che ricevendo la tabella e il giocatore, verifica se quel giocatore ha vinto (controllo solo in orizzontale e verticale)

```
function  vince = controllaSeVince(A , idGiocatore)
```

Gruppo 3:

Sviluppare il **Main** e la funzione **CONTROLLASEPIENA** che ricevendo la tabella verifica se la tabella è piena e quindi se vi è la possibilità di fare altre giocate.

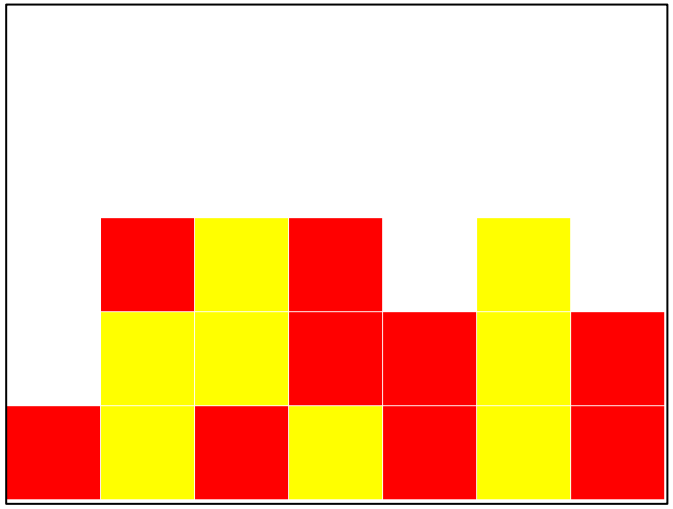
```
function [flag] = controllaSePiena(A)
```

Forza 4 con schema grafico

Scrivere una funzione ‘schema_grafico’ che riceve in input la matrice con la tabella di forza quattro e genera lo schema di gioco di Forza quattro (vedi immagine di esempio), in particolare:

- La tabella di gioco è una matrice 6 righe e 7 colonne
- Ogni pedina è un quadrato di lato 101
- Impostare la pedina del giocatore 1 di colore rosso mentre quella del giocatore 2 di giallo. Le caselle vuote restano bianche bianco.

P.S.
La matrice che la funzione riceve in input è una 6x7 dove ogni cella corrisponde ad un quadrato dello schema grafico in output. Gli zeri della matrice corrispondono a casella vuota, 1 corrisponde a pedina del giocatore 1 e 2 corrisponde a pedina del giocatore 2.



https://www.youtube.com/watch?v=u4pVGN0aHMs&ab_channel=TuttoSpot80

P2: interpretazione e modifica del codice

```
clear
anno=2019;
stringa='Unità_Italia_17_Marzo_1861';
stringa2='';
for x=1:length(stringa)
    if stringa(x)>=48 && stringa(x)<=57
            stringa2=[stringa2 stringa(x)];
    end
end
anno_ui = str2double(stringa2(3:6));
n_anni_italia = anno-anno_ui;
disp(stringa2) %0
disp(['n_anni_italia = ', n_anni_italia]) %1
disp(['n_anni_italia = ', num2str(n_anni_italia)]) %2
whos %3
```

P1: interpretazione e modifica del codice

HINT: nella tabella ASCII lo ‘0’ corrisponde al 48.

HINT: *str2double* converte una sequenza di caratteri passata in ingresso in un numero

HINT: *whos* stampa a schermo una tabella con le seguenti informazioni per ogni variabile

Name Size Bytes Class

- Si dica se l’istruzione %1 è corretta. Perché?
- Si dica se l’istruzione %2 è corretta. Perché?
- Si riporti quello che viene stampato a schermo durante l’esecuzione del programma, considerando solo le istruzioni ritenute corrette.