

LEZIONE:

Istruzioni Condizionali

Tempo della lezione: 45-60 Minuti. - Tempo di preparazione: 2 Minuti.

Obiettivo Principale: Questa lezione introduce il concetto di istruzione condizionale e le specifiche istruzioni “se” e “se-altrimenti”.

SOMMARIO:

Questo è un gioco per l'intera classe. Dopo aver introdotto in generale il concetto di istruzione condizionale ed aver spiegato le specifiche istruzioni 'se' e 'se-altrimenti', gli studenti si esercitano tutti insieme usando un programma già scritto, un mazzo di carte, e un po' di fortuna

OBIETTIVI:

Gli studenti:

- Acquisiscono esperienza nel determinare il risultato (vero o falso) di condizioni logiche.
- Valutano delle condizioni logiche per determinare quale ramo di un programma seguire.

MATERIALI:

- 1 mazzo di carte da gioco francesi.

PREPARAZIONE:

Separa le carte vestite (figure) dalle altre. I Jolly vanno tenuti assieme alle carte non vestite (numerate), e possono essere utilizzati per concedere un vantaggio

speciale a chi li detiene, come ad esempio la vittoria istantanea o la cancellazione di tutti i punti dell'avversario.

Proietta sullo schermo il programma appropriato al livello della tua classe (nell'ultima pagina ci sono degli esempi).

VOCABOLARIO:

Istruzione Condizionale – Un'istruzione che richiede la verifica di una condizione

Istruzione “se” – Un'istruzione che determina se eseguire o meno altre istruzioni

Altrimenti – Un altro modo per dire “in caso contrario”.

Funzione – Una parte di codice che può essere chiamata più volte.

Incrementare – Aggiungere una certa quantità (solitamente 1).

Decrementare – Sottrarre una certa quantità (solitamente 1).

Istruzioni annidate – Istruzioni all'interno di altre istruzioni.

RIPASSO:

Questa sezione di ripasso ha lo scopo di far ricordare agli studenti la precedente lezione. Se stai svolgendo le attività in un ordine diverso, sostituiscila con il ripasso degli argomenti svolti.

Le domande che seguono suppongono che questa sia la lezione 10 del percorso completo (interattivo + senza rete).

Domande per la discussione di classe:

- Cosa abbiamo fatto nella scorsa lezione?
- Che differenza c'è tra un blocco "mentre" ed un blocco "se"?

Domande per la discussione tra compagni di banco:

- Riuscite a pensare a qualcosa che facciamo tutti i giorni che possa essere descritto con un blocco "mentre"?



Le istruzioni condizionali sono il modo con cui i calcolatori prendono le decisioni.

INTRODUCI:

Questo gioco può essere rifatto più e più volte.

È infinitamente personalizzabile, con variazioni a seconda dell'età, dell'esperienza e del numero di studenti.

In questa lezione la descrizione è orientata principalmente alle classi dalla terza alla quinta, ma la puoi adattare a tuo piacimento a classi di altro livello mantenendone inalterato lo spirito.

Per cominciare, gli studenti avranno bisogno di un ripasso sulle istruzioni condizionali, anche se le hanno già viste nelle attività interattive con Blockly. Può essere utile iniziare con qualche esempio pratico che serva da stimolo:

“Se tutti voi alzerete la mano, allora applaudirò. In caso contrario, mi toccherò il naso.”

Dopodiché, toccati il naso fino a quando tutta la classe non alza la mano, quindi applaudi per far capire che hanno tutti verificato la condizione.

“Che cosa è appena successo?”

Gli studenti, probabilmente, proveranno a descrivere gli eventi passo per passo, e starà a te guidarli nella direzione voluta.

“Vi ho dato una ‘condizione’ giusto? Ho detto che se si fosse verificata una certa condizione avrei fatto una cosa, ma se la condizione non si fosse verificata avrei fatto qualcos'altro. Le istruzioni condizionali sono il modo in cui i calcolatori prendono le decisioni.”

Dopodiché puoi provare di nuovo. Questa volta introduci il termine “altrimenti”,

“Se tutti voi mettete le mani sui vostri banchi, mi gratterò la testa.
Altrimenti metterò la mano sul fianco.”

Questo caso è abbastanza simile al primo e gli studenti dovrebbero essere in grado di intuire il significato di “altrimenti”, ma se hanno delle difficoltà, procedi spiegando che ha un significato simile a “in caso contrario”.

“Ottimo. Rendiamo il tutto un po' più interessante. Ho qui un mazzo di carte e stiamo per fare un gioco.

Se pescò una carta rossa, faccio un punto.
Altrimenti, voi fate un punto.”

Scrivi queste semplici istruzioni sulla lavagna in modo da poter indicare i vari casi quando si presentano.

Pesca qualche carta dalla cima del mazzo e assegna i punti di conseguenza. Quando la carta è rossa fai vedere quale caso consideri (quello dopo il “se”) e quando è nera fai notare che corrisponde al caso “altrimenti”.

Dopo qualche turno (o quando gli studenti hanno tutto chiaro) fai terminare il gioco e preparati ad innalzare la difficoltà.

“Va bene, ora useremo questa idea delle condizioni, o ‘istruzioni condizionali’ per fare un gioco. Chiamiamo ‘condizionali’ queste istruzioni ‘se’ perché contengono una condizione. Qualcosa è vero oppure no. Se è vero eseguiamo le istruzioni nel ramo ‘se’. Altrimenti eseguiamo le istruzioni nel ramo ‘altrimenti’.”

Per far sì che le regole siano chiare a tutti, scrivile sulla lavagna. Il codice seguente rappresenta il modo in cui le istruzioni condizionali sono scritte in Javascript, uno dei linguaggi di programmazione più usati al mondo.

Nelle attività online di questo corso, abbiamo scritto del codice con Blockly, che utilizza dei blocchi grafici per rappresentare il codice. Il codice che abbiamo scritto in queste attività può essere rappresentato anche in altri linguaggi di programmazione. Per esempio, nelle nostre attività online, se cliccate “Mostra il Codice” nell’interfaccia viola, potete vedere i blocchi dell’area di lavoro rappresentati in Javascript.

Di seguito viene fornito il codice dell’attività che svolgeremo oggi.

Quando leggete il codice seguente tenete presente che la parola “if” indica “se” e che la parola “else” indica “altrimenti”. Poiché il linguaggio in cui è scritto questo codice è esattamente quello che usano i calcolatori abbiamo volutamente lasciato questi termini in inglese.

```
1) if (carta.colore == rosso) {  
2) squadra.punti += 1;  
3) }  
4) else {  
5) squadra.punti -=1;  
6) }
```

Probabilmente sarà la prima volta che gli studenti si trovano di fronte ad un codice del genere, e quindi dovrai discutere questo codice linea per linea:

1) Questa linea è l’istruzione “se”. La parola “if” significa “se”. Questa linea vi permette di decidere che “Se il colore della carta è uguale a rosso, fai QUALCOSA”. Notate che ci sono due segni “uguale”? In molti linguaggi di programmazione questo è il modo per dire che stiamo solo controllando l’uguaglianza fra due cose. Questa linea finisce con una “parentesi graffa aperta”. È lì per farci sapere che la linea successiva appartiene al blocco “se”, che eseguiamo se la condizione è vera.

2) Questa linea dice letteralmente “punti più uguale uno”. Significa che dovete prendere il numero di punti che avete e aggiungerci uno. Qualsiasi sia il risultato, quello è il vostro nuovo numero di punti.

3) Questa è una “parentesi graffa chiusa”. È qui per dirvi che il blocco “se” è finito.

4) Ecco l’inizio del blocco “altrimenti”. Ricordate che “else” significa proprio “altrimenti”. Alla fine della linea vedrete un’altra parentesi graffa aperta. Ciò significa che questo è l’inizio del blocco che eseguiamo se la condizione non è vera.

5) Questa linea è molto simile alla linea 2 ma con una piccola differenza. Riuscite a vederla? Che cosa fa secondo voi? Significa che dovete prendere il numero di punti che avete e sottrarre uno. Qualsiasi sia il risultato, questo è il vostro nuovo numero di punti.

6) Infine, alla riga 6 abbiamo l’ultima parentesi graffa chiusa. Ci dice che il blocco “altrimenti” è terminato.

Ecco il Javascript scritto accanto allo stesso codice, rappresentato in pseudocodice: un codice scritto in maniera informale in modo che sia facile da leggere. Ogni linea dello pseudocodice spiega che cosa fanno le linee in Javascript.

Javascript	Pseudocodice
if (carta.colore == rosso) { squadra.punti += 1; }	Se il colore della carta è nero, allora il numero di punti della squadra aumenta di 1
else { squadra.punti -=1; }	Altrimenti i punti della squadra diminuiscono di uno.

Dopo la spiegazione, è utile fare un passo indietro e guardare la situazione complessiva. **Che cosa vuol dire tutto questo? Possiamo spiegarlo a parole? Sostanzialmente dice che se una persona pesca una carta nera, la squadra riceve un punto. In caso contrario la squadra perde un punto. Iniziamo?**

Scegli otto volontari, quattro per ogni squadra, che saranno i “programmatori” (Assumeremo che il gioco venga giocato con almeno 20 studenti; se ne hai di meno controlla gli adattamenti alla fine di questa lezione).

Fai pescare una carta vestita ad ogni programmatore. Questo non è necessario ma è una maniera divertente per scegliere quale squadra inizierà per prima (utilizza le regole alla fine della lezione). Scegli uno studente non-programmatore per prendere le carte numerate e distribuisce una ciascuna, a faccia in giù, a tutti gli studenti non-programmatori. Questo permette a tutti di sentirsi coinvolti anche se dovessi aver tempo solo per pochi turni.

Fai scegliere al primo programmatore un qualunque studente con una carta numerata. Quello studente si alzerà e mostrerà a tutti la sua carta. **Di che colore è? A che linea corrisponde? Che cosa succede ai punti della squadra di quel programmatore?**

Prosegui con un programmatore della squadra opposta e ripeti fino a quando tutti i programmatori non hanno preso parte ad un turno. **Quanti sono i punti finali? Quale squadra ha vinto?**

Se il gioco si è svolto abbastanza velocemente, scegli un altro programma e ricomincia. Se è andato **davvero** molto bene, sfida gli alunni ad aiutarti a modificare il programma per avere un maggior numero di opzioni. Dopo averci giocato un paio di volte, prova ad espanderlo e ad aggiungere più caratteristiche come “blocchi mentre” o “istruzioni annidate”.

ADATTAMENTI:**CLASSE**

Per studenti di prima e seconda elementare: Rappresentare graficamente il tuo “programma” aiuta gli studenti di questa età. Ti suggeriamo di evitare di scrivere programmi Javascript. In ultima pagina trovi una versione grafica del programma Javascript discusso nel testo precedente. Usa algoritmi semplici e cambia le cose solo dopo che gli studenti si sono abituati bene al funzionamento delle regole. L'obiettivo principale per questa fascia di età è insegnare la logica piuttosto che come leggere un programma.

Per studenti di terza, quarta e quinta elementare: Questa fascia dovrebbe lavorare per lo più come descritto nella lezione. Incrementa gradualmente la complessità del programma.

Per studenti delle scuole medie: Gli studenti possono lavorare in gruppi più piccoli (4-8) e competere gli uni contro gli altri. Si possono pescare le carte dal mazzo, piuttosto che indicare gli altri studenti. Fai aumentare la difficoltà dei programmi a discrezione degli studenti. Incoraggia gli studenti a creare i propri programmi.

ESPERIENZA

Principiante: Prova ad usare una sola istruzione condizionale alla volta. Potrebbe essere d'aiuto scrivere i programmi con immagini o diagrammi piuttosto che in Javascript.

Intermedio: Spiega agli studenti il codice scritto in Javascript, in modo che possano avere un assaggio di come sia la programmazione testuale. Se sono a loro agio con gli altri concetti di Javascript, considera di usare delle strutture ad oggetti come `studente.carta`, `carta.colore` o spingiti oltre usando istruzioni annidate ed, eventualmente, anche funzioni: `carta.valore.perDue()`.

Avanzato: Spendi meno tempo a lavorare con l'intera classe e più tempo a lavorare in piccoli gruppi.

Fai scrivere il programma agli studenti e sottoponigli delle sfide come:

- Crea un programma in cui pescare un asso nero raddoppia il tuo punteggio.
- Crea un programma dove ci sia un risultato diverso a seconda che la carta sia pari piuttosto che dispari oppure nera piuttosto che rossa.

NUMERO DI GIOCATORI

1: Questa situazione può essere vista come un solitario. Gli studenti possono pescare da un mazzo e confrontare le carte con il programma di un solitario già esistente, o con uno che hanno creato da soli. Un esempio potrebbe essere quello di far eseguire allo studente un programma che richieda di ottenere un certo punteggio entro un certo numero di mosse per “vincere”. Un altro esempio potrebbe far giocare lo studente contro una “macchina”, dove i punti possono essere accumulati da due entità differenti ed il vincitore è quella che ha totalizzato più punti alla fine del gioco.

2-6 (un numero pari di studenti): Va considerato come un gruppo di piccole dimensioni. Di solito gli studenti completano i turni così velocemente che cercare di coinvolgere tutti distribuendo una carta per ogni banco è una distrazione più che un beneficio. Dividi gli studenti in due squadre e posiziona un mazzo di carte al centro. Fai pescare gli studenti a turno. Non c'è bisogno di scegliere dei “programmatori” volontari per questa variante dal momento che tutti gli studenti pescheranno le carte.

3-7 (un numero dispari di studenti): Gioca come descritto sopra, ma con l'aggiunta di un “Contatore del Programma” che tiene il punteggio e prende la decisione finale su quali linee di codice sono pertinenti.

8+: Per questa dimensione il risultato migliore si ottiene lavorando come per l'intera classe. Se gli studenti sono più grandi, fagli pescare da un mazzo al centro della stanza, piuttosto che far loro scegliere una carta sul banco di qualcun altro.

REGOLE:

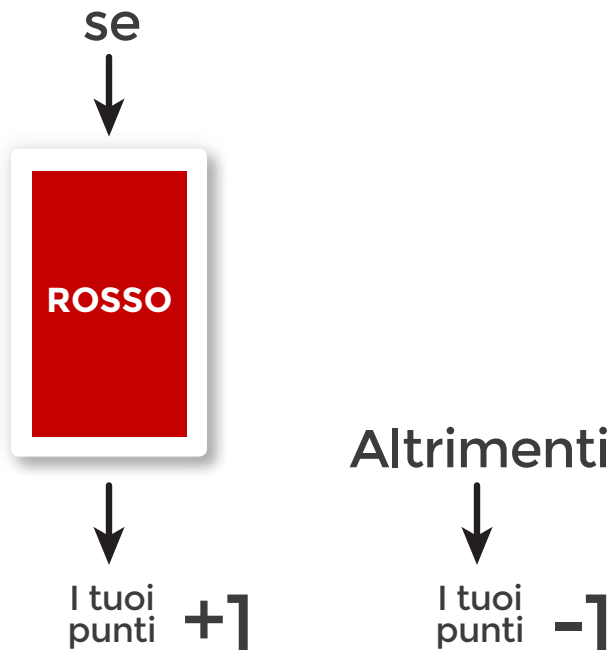
Scegli 8 volontari (4 per ogni squadra) che saranno i “programmatori”.

Fai pescare i programmatori dal mazzo con le carte vestite per decidere chi inizierà per primo.

L'ordine di gioco è il seguente: Re (K), Regina (Q), Fante (J). A parità di figura vale il seme, in quest'ordine: Cuori, Quadri, Fiori, Picche.

Esempio:

- Re di Cuori
- Re di Quadri
- Re di Fiori
- Re di Picche
- Regina di Cuori
- Regina di Quadri
- Regina di Fiori
- Regina di Picche
- Fante di Cuori
- Fante di Quadri
- Fante di Fiori
- Fante di Picche



<-- Programma di esempio per studenti di prima, seconda o terza elementare, oppure per studenti di livello principiante

Programma di esempio:

Per studenti di quarta o quinta elementare, oppure di Livello intermedio

```
while (carta.valore < 5){
    punti ++;
}
// Questo programma permette al giocatore di
// pescare una carta "mentre" (cioè "per tutto il tempo
// in cui") la carta scelta è minore di 5

// Ogni volta che il giocatore pesca una carta riceve
// un punto, indipendentemente dal valore
```

Programma di esempio:

Per studenti delle scuole medie, oppure di Livello Esperto

```
while (carta.valore < 5){
    squadra.punti ++;

    if (carta.valore == 3){
        squadra.punti = squadra.punti + 2;

        if (carta.seme == "picche"){
            altraSquadra.punti += 2;
        }
    }
}

// Questo programma permette al un giocatore di
// pescare una carta "mentre" (cioè "per tutto il tempo
// in cui") la carta scelta è minore di 5

// Ogni volta che il giocatore pesca una carta riceve
// un punto, indipendentemente dal valore. Se la carta
// scelta è un 3 la squadra riceve altri 2 punti, e se il
// suo seme è "picche" anche l'altra squadra riceve 2
// punti
```