**I.T.T.S.**

**“S.Fedi-E.Fermi”**

Informatica

2017/2018

3 IB

*Vestri Luigi*

*Caramelli David*

**Goosegame**

**Bujor Marcel Iulian**

**Galligani Stefano**

**24/01/18-07/03/18**

# SOMMARIO

[**SOMMARIO**](#_ebh6of1v29mm) **2**

[**GOOSEGAME**](#_y8n4eupc7w5z) **3**

[Scopo](#_xjdlz2jlrv4q) 4

[Analisi Funzionale](#_n9ah1tgyh7sc) 4

[Analisi Tecnica](#_a6w0h7g19n1g) 4

[Manuale per l’utente](#_6ebexjap4r0j) 4

[Obiettivo](#_7wh0q4lkz9q) 4

[Avvio](#_7nl7yri5lrr5) 4

[Screenshot del programma in esecuzione](#_usxiuvh07q7t) 5

[Requisiti Tecnici](#_v8vp0cd5cfw4) 5

[Test data set\Debug](#_e1w30ygeb02o) 5

[Bibliografia\Sitografia di riferimento](#_v5q4s8af1wb8) 5

[Osservazioni e conclusioni personali](#_82d27ag9vihw) 5

# GOOSEGAME

Il gioco dell’oca è un tipo di gioco da tavolo tradizionale per bambini. Si tratta di un gioco

estremamente semplice, in cui il vincitore è determinato esclusivamente dalla sorte. Si

gioca su tabellone sul quali è disegnato un percorso a spirale, generalmente sinistrorsa,

composto da 63 caselle, contrassegnate con numeri o altri simboli. I giocatori iniziano con

un segnalino nella casella di partenza e, a turno, procedono lungo il percorso di un

numero di caselle ottenuto attraverso il lancio di una coppia di dadi. Lo scopo del gioco è

raggiungere la casella centrale

della spirale.

Alcune caselle di arrivo hanno un effetto speciale. Nella versione tradizionale, le caselle

che rappresentano oche consentono di spostarsi subito in avanti di un numero di caselle

pari a quelle coperte dal movimento appena effettuato. Queste caselle sono collocate ogni

nove caselle a partire dalle caselle 5 e 9 (una conseguenza di questa disposizione è che

un lancio iniziale di 9 porta immediatamente il giocatore alla casella 63 e quindi alla

vittoria).

Le altre caselle speciali sono le seguenti:

alla casella 6 (“il ponte”) si paga la posta e si ripete il movimento come nelle caselle

con le oche;

alla casella 19 (“casa” o “locanda”)nella casella un altro segnalino, che viene a sua volta “imprigionato”;

alla casella 42 (“labirinto”) si paga la posta e si torna alla 39;

alla casella 58 (“scheletro”) si paga la posta e si torna alla 1.

La casella d’arrivo (la 63) deve essere raggiunta con un lancio di dadi esatto;

altrimenti, giunti in fondo, si retrocede dei punti in eccesso.

## Scopo

Realizzare un programma che consenta di effettuare n gare per k giocatori.

La lista dei vincitori generale è stilata sulla base del numero di dadi tirati, e del numero di partite vinte.

## Analisi Funzionale

Nel gioco saranno presenti:

un menù, per leggere il regolamento ed eventualmente rimuovere i suoni;

la possibilità di scegliere il nome e la pedina per ogni giocatore;

un tabellone, su cui saranno mostrate le pedine (verdi), le caselle e le caselle speciali (ognuna di un colore diverso);

un elenco delle posizioni dei giocatori;

un elenco degli eventi (ultimi tiri del dado, giocatori che restano fermi ecc.);

un’immagine relativa alla casella speciale raggiunta da un giocatore;

una traccia musicale relativa alla casella speciale raggiunta dal giocatore;

una traccia musicale di sottofondo che inizierà appena sarà fatto il cout del tabellone e si interromperà soltanto quando arrivati su una casella speciale parte la traccia relativa per ricominciare appena finita;

i titoli di coda con relativa traccia musicale, contenenti anche la classifica.

Ai fini di una migliore fruizione del gioco, è stato applicato un vincolo aggiuntivo rispetto al testo del problema, cioè: quando un giocatore ottiene 9 al primo turno, invece di passare per tutte le caselle delle oche ed arrivare subito alla fine, andrà alla casella 26 se ha ottenuto un 3 e un 6, alla 53 se ha ottenuto un 4 e un 5.

## Analisi Tecnica

Legenda colori:

nero=scritta normale

rosso=linguaggio c/c++

* main:

Ci sono tre cicli principali:

il più interno, che regola i turni, facendoli svolgere normalmente se il giocatore non è fermo a causa di una casella speciale;

un ciclo intermedio, do{}while, che controlla se un giocatore è arrivato alla casella finale;

il ciclo più esterno, che fa ricominciare la partita se ce ne sono ancora altre da fare.

* menuIniziale:

Qui, l’utente ha la possibilità di disabilitare i suoni tramite una variabile booleana, leggere il regolamento o avviare la partita.

* cinTotP :

All’utente viene chiesto quante partite ha intenzione di giocare. Non c’è un limite effettivo al massimo di partite selezionabili.

* inputGiocatori:

Inizialmente verrà chiesto, attraverso due do{}while il numero di giocatori e di bot, dopo di che attraverso un for e uno switch viene attribuito il nome del bot, da bot2 a bot5.

In seguito attraverso un for verranno chiesti i nomi dei giocatori e le pedine, che verranno applicate attraverso uno switch, un giocatore non potrà scegliere la pedina che ha scelto precedentemente un altro se no verrà visualizzato un messaggio di errore e verrà chiesto di inserirla nuovamente.

Con un altro for e uno switch verranno attribuite le pedine ai bot, tra quelle restanti.

* coutTabellone:

Questa funzione è in realtà un insieme di altre funzioni, che collaborano per stampare a schermo il tabellone di gioco.

* coutPos:

Attraverso un for verranno visualizzate in alto a destra le posizioni attuali dei vari giocatori, individuati dal proprio nome.

* creaQuad:

Questa funzione crea una casella in una posizione che dipende da un doppio ciclo for(){ presente in coutTabellone. Essa può creare un totale di 110 caselle, soggette però a un controllo che ne farà stampare solo 63.

* smussaAngoli:

Questa funzione ha lo scopo di sistemare alcune imperfezioni nel tabellone completo, dovute principalmente a bordi e angoli.

* tiroDado:

Attraverso la funzione rand vengono estratti due numeri casuali, i quali vengono visualizzati in due caselle una accanto all'altra, creati attraverso diversi for e putchar, funzione che sostituisce ai valori numerici il corrispettivo ascii.

Prima che nella casella siano visualizzati i numeri effettivi dei dadi, con un for e due if vengono visualizzati, all’interno dei quadrati che simbolizzano i dadi, altri numeri casuali che scorrono dall'alto in basso, con un movimento a velocità decrescente creato da uno Sleep, creando un effetto simile a quello delle slot machine.

Infine, una volta visualizzati il numero uscito sui dadi, viene fatta la somma dei due dadi e questo valore viene passato al main.

* calcoloPosizione:

Questa funzione calcola la posizione del giocatore che ha appena tirato il dado, controllando anche se si trova su una casella speciale (e in tal caso riprodurre un frammento di canzone corrispondente) o se ha superato la casella 63.

* calcoloVincitori:

Questa funzione controlla i punteggi dei vari giocatori e, tramite alcuni vettori di appoggio e un funzionamento abbastanza complesso da essere poco comprensibile persino al programmatore, riesce a calcolare la classifica dei giocatori e salvarla in un vettore, che verrà poi usato nei titoli di coda.

* coutValore:

Questa funzione controlla, casella per casella, se è presente un giocatore, e in tal caso stampa la rispettiva pedina, altrimenti stampa il numero della casella, eventualmente colorato in caso si tratti di una casella speciale.

* coutLog:

Attraverso uno switch vengono visualizzati, a destra sotto la posizione dei giocatori, gli ultimi spostamenti dei vari giocatori e inoltre attraverso una serie di if se una delle ultime azioni è stata su una casella speciale viene descritto quello che è successo in quella casella, insieme a eventuali ripercussioni sul futuro(es. resti fermo 3 turni).

Arrivata alla ventesima riga di storia il log si cancella e riparte dalla cima.

* clearLog:

Cancella tutte le righe della storia o i disegni attraverso un for.

* disOca/Casa/Pozzo/Prigione/Scheletro:

Sono i vari disegni riferiti alle caselle speciali(oca, casa, pozzo, prigione e scheletro), attraverso una lunga serie di putch.

* coda:

Questa funzione fa scorrere i titoli di coda, in cui è compresa anche la classifica (idea basata sul celebre videogioco Left 4 Dead) tramite una matrice di caratteri (o vettore di stringhe) che varia ad ogni ciclo, copiando ogni stringa nella precedente (cioè quella soprastante) o, nell’ultima riga, una frase presa da una matrice secondaria, che contiene le stringhe che dovranno essere stampate.

I titoli di coda durano 20 secondi, come la traccia musicale scelta.

## Manuale per l’utente

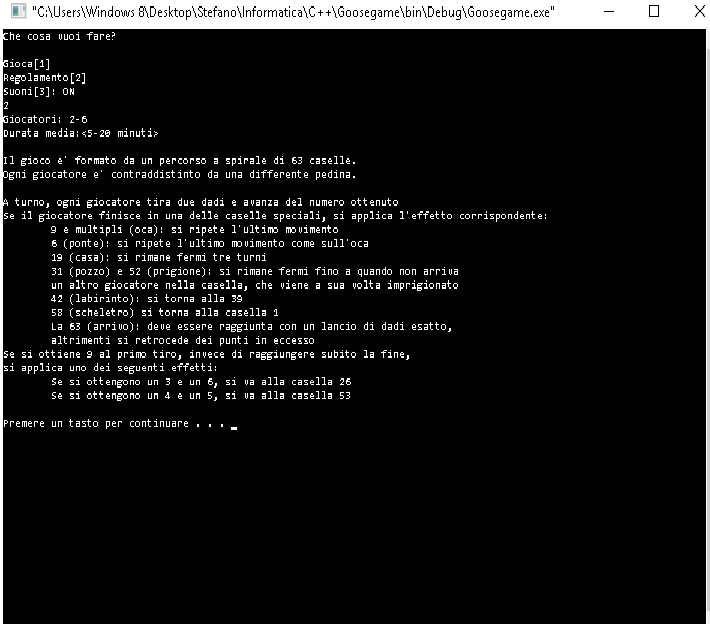
### Obiettivo

Divertirsi con una versione digitale del gioco dell’oca con gli amici o, eventualmente, con i bot

### Avvio

Basterà avviare codeblocks e cliccare su build e run, o aprire il file di gioco dalla shell. In seguito sarete guidati passo per passo da un'interfaccia molto facile ma concisa che vi spiegherà come continuare.

### Screenshot



### 

### Flusso del programma

Come prima cosa verrà visualizzato un menu iniziale con la possibilità di rimuovere i suoni, vedere le regole o iniziare il gioco.

Una volta scelta l’opzione per iniziare il gioco, verrà chiesto di inserire il numero di partite che si vogliono giocare, il numero di giocatori\bot con relativo nome e una pedina scelta tra 6 possibili, pedina e nome dei bot saranno inseriti in automatico.

Dopo aver visualizzato il tabellone verranno tirati due dadi attraverso la pressione di un qualsiasi tasto.

Verrà fatta la somma dei due dadi e la pedina del giocatore\bot, sarà mossa nella rispettiva casella, con eventuali modifiche causate dalle caselle speciali.

Nel corso del gioco saranno mostrati sulla destra la posizione di ogni giocatore\bot e una breve storia con quello che è successo negli ultimi passi.

Dopo che un giocatore\bot è arrivato alla casella 63 con un colpo preciso di dadi verrà introdotto tra i vincitori e si proseguirà alla successiva partita; dopo aver giocato più partite verrà mostrata la classifica dei vincitori, se due giocatori hanno fatto lo stesso numero di vittorie si guarderà il numero di volte che è stato lanciato il dado per determinare il vincitore in assoluto. Infine verranno visualizzati i titoli di coda, con le funzioni e chi le ha create.

### Requisiti Tecnici

* Sistema operativo: Windows (preferibilmente 10)
* 1 MB di RAM
* Memoria: 20 MB di spazio disponibile

### Test data set\Debug

Dopo aver installato e avviato il programma, il tabellone potrebbe avere i lati di alcune caselle in posizione errata. Per risolvere questo problema basta cercare la funzione creaQuad nel codice sorgente e scrivere, al posto della funzione di output “cout”, la seguente riga di codice:

cout << “ ”;

con quattro spazi anziché la tabulazione.

In alcuni momenti si potranno avere dei problemi con la visualizzazione del log, che però si aggiusterà dopo pochi secondi.

Infine a volte il cout tabellone non viene fatto immediato ma in maniera sequenziale, anche se raramente.

## Bibliografia\Sitografia di riferimento

<https://www.youtube.com/watch?v=sbn-7WwXMvo> (tutorial per riprodurre un suono in C++)

Corso di Informatica di Fiorenzo Formichi e Giorgio Meini Zanichelli editori

Metodo selection sort per l’ordinamento di un vettore

## Osservazioni e conclusioni personali

E’ stato molto interessante lavorare per il primo gioco, specialmente perché è stato molto piacevole lavorare in gruppo e riuscire a creare un programma tanto ben riuscito.

E’ stato anche interessante notare come, a partire da un insieme di istruzioni basilari e dei caratteri, si possa riuscire a creare un videogioco che possa intrattenere e divertire gli utenti.