**Snake And Ladder Game**

1. **Overview**

Game Ular tangga adalah boardgame yang dapat dimainkan oleh dua orang atau lebih. Dalam game ini terdapat sebuah papan yang berbentuk bujur sangkar. Dalam papan tersebut terdapat kotak – kotak yang berjumlah seratus. Kotak – kotak tersebutlah yang menjadi jalan yang digunakan oleh pemain.

1. **Rules**

Setiap pemain memiliki maskotnya masing – masing. Dalam game ini menggunakan satu buah dadu yang digunakan untuk menentukan berapa banyak langkah yang digunakan pemain untuk bergerak. Dalam papan terdapat kotak – kotak berjumlah seratus. Angka pertama terdapat di posisi kiri bawah dari papan dan menjadi tempat permulaan dari permainan. Angka tertinggi atau seratus berada di sebelah kiri atas dari papan dan menjadi tempat berakhirnya permainan.

Pemain akan bergerak sebanyak angka yang keluar dari dadu. Permainan akan berjalan kekanan, kemudian jika pemain berada pada angka 10 maka pemain akan bergerak ke atas lalu bergerak kekiri. Apabila pemain berada di angka 20 maka permainan akan bergerak keatas kemudian kekanan. Hal tersebut akan berlaku sama setiap selisih 10 angka.

Dalam game ini terdapat obstacle – obstacle yang berupa tangga dan ular. Jika pemain berada di tangga bagian bawah maka pemain akan bergerak menaiki tangga menuju bagian atas tangga. Jika pemain mengenai mulut ular maka pemain akan turun ke ekor ular. Jika pemain tepat berada diatas kotak ke seratus maka pemain dinyatakan menang.

1. **Fungsi – Fungsi dan class**
2. Display

Fungsi ini digunakan untuk menampilkan semua variable variabel dan fungsi yang ada agar user dapat melihat dan memainkan gamenya.

1. Pembuat papan

Pada fungsi ini terdapat variabel array yang digunakan untuk membuat papan. Papan tersebut terbuat dari beberapa fungsi looping yang membuat kotak dalam papan.

1. Dadu

Fungsi ini digunakan untuk membuat dadu dan didalam fungsi ini juga terdapat fungsi roll untuk mengacak dadu.

1. Main

Fungsi adalah fungsi induk yang berguna untuk memanggil semua fungsi yang ada pada project.

1. **Relasi Antar Kelas**
2. **Inheritance**

* Class obstacle dan class obstacleStair

Class obstacle adalah class utama. Class ini memiliki variabel – variabel sendiri. Dalam class ini terdapat variabel startPost yang berfungsi untuk menentukan titik awal posisi. Kemudian juga terdapat variabel endPost yang berfungsi untuk menentukan titik akhir. Class obstacle ini menurunkan fungsi – fungsinya terhadap class obstacleStair. Class obstacleStair ini adalah sebuah class yang berguna untuk membuat mascot pemain bergerak menaiki tangga menuju atas atau kotak yang dituju. Pada class obstacleStair ini mengambil fungsi atau variabel dari class utama. Class yang diambil adalah variabel startPost dan variabel endPost. Fungsi variabel tersebut sama dengan fungsi yang ada pada class obstacle hanya saja pada class turunannya variabel ini memiliki value yang berbeda sesuai dengan tempat tangga berada.

* Class obstacle dan class obstacleSnake

Class obstacle adalah class utama. Class ini memiliki variabel – variabel sendiri. Dalam class ini terdapat variabel startPost yang berfungsi untuk menentukan titik awal posisi. Kemudian juga terdapat variabel endPost yang berfungsi untuk menentukan titik akhir. Class obstacle ini menurunkan fungsi – fungsinya terhadap class obstacleSnake. Class obstacleSnake ini adalah sebuah class yang berguna untuk membuat mascot pemain bergerak menuruni tubuh ular menuju kotak yang ada ekor dari ular. Pada class obstacleSnake ini mengambil fungsi atau variabel dari class utama. Class yang diambil adalah variabel startPost dan variabel endPost. Fungsi variabel tersebut sama dengan fungsi yang ada pada class obstacle hanya saja pada class turunannya variabel ini memiliki value yang berbeda sesuai dengan mulut ular berada.

* Class mascot

Class ini berguna untuk membuat sebuah mascor. Class mascot yang memiliki variabel – variabel dan fungsi yang mengatur mascot. Pada class mascot terdapat variabel int Id, bool selection, int moveChar, int currentPost. Kemudian class tersebut diturunkan untuk menjadikan pemain yang lain.

1. **Aggregation**

* Hubungan class dadu dan class mascot

Class dadu adalah class yang berfungsi untuk melakukan roll agar mendapatkan angka tertentu. Dari fungsi roll dari dadu, user akan mendapatkan value. Fungsi roll tersebut nantinya akan dipakai oleh class mascot. Hal yang diambil dari fungsi roll tersebut adalah value. Value tersebut nantinya akan digunakan untuk oleh mascot untuk bergerak dan juga sebagai batasan gerak dari mascot.

* Hubungan class papan dan class mascot

Pada class papan terdapat fungsi pembuat kotak – kotak pada papan. Pada class ini juga terdapat sebuah fungsi yang berguna untuk memberikan rules permainan. Rules tersebut adalah jika mascot berada pada kotak ke 10 maka ia akan bergerak ke atas lalu ke kiri dan apabila mascot berada pada kotak ke 20 maka mascot akan bergerak ke atas kemudian ke kanan. Lalu fungsi fungsi tersebut dipakai atau diterapkan pada class mascot untuk membatasi gerak dari mascot.

1. **Composition**

Pada hal ini class papan, class dadu dan class mascot sangat diperlukan satu sama lain. Jika tidak ada salah satu maka permainan akan tidak bisa berjalan. Mascot membutuhkan dadu untuk bergerak. Mascot membutuhkan papan sebagai tempat bergerak.

* Class papan dan class mascot

Class papan adalah class yang berfungsi untuk membuat sebuah papan yang digunakan oleh mascot untuk bergerak. Jika class papan ini tidak ada, maka dapat dipastikan permainan tidak bisa dimainkan. Hal ini terjadi karena mascot tidak memiliki tempar untuk bergerak atau arah dari gerak mascot.

* Class dadu dan class mascot

Class dadu adalah class yang menghandle value untuk menggerakkan mascot. Jika class dadu tidak ada, maka mascot tidak bisa bergerak dan permainan akan tidak berjalan sebagaimana mestinya. Hal ini terjadi karena value yang dibutuhkan oleh mascot untuk bergerak itu tidak ada.

* Class papan dan class dadu

Kedua class ini sangat berpengaruh satu sama lain. Class dadu yang menentukan value kemudian akan menggerakkan mascot pada papan. Value tersebut disingkronisasi dengan jumlah kotak yang dibutuhkan sebanyak value dari dadu.