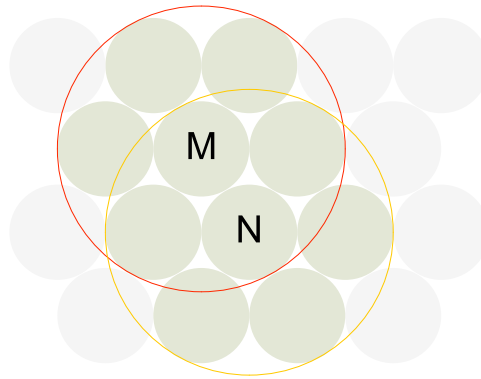


BAB VI KESIMPULAN DAN SARAN

1. Kesimpulan

Berikut beberapa hal yang dapat disimpulkan setelah mengerjakan Tugas Akhir :

1. *Game* ini berhasil dibuat dengan mengimplementasikan *Object Oriented Programming*. *Game* ini dibuat dengan menggunakan 13 *class* (pada *game single player*) dan 24 *class* (pada *game multiplayer*).
2. Pencarian semua *bubble* yang berwarna sama untuk diledakkan dilakukan berdasarkan pola tertentu, yaitu pola normal dan pola menyorok.



Keterangan :

M = *bubble* dengan pola menyorok.

N = *bubble* dengan pola normal.

Tetangga *bubble* M adalah *bubble* yang berada dalam lingkaran garis merah, sedangkan tetangga *bubble* N adalah *bubble* berada dalam lingkaran garis kuning. Tampak pada gambar di atas bahwa dalam melakukan pencarian tetangga, baik pola normal maupun pola menyorok, pola pencarian tetangga adalah sama. Tetangga yang dicari adalah *bubble* yang berwarna sama dengan *bubble* yang ditembakkan (dalam hal ini *bubble* yang ditembakkan adalah *bubble* M dan N). Tetangga *bubble* M dan *bubble* N yang telah ditemukan juga akan dicari lagi tetangganya. *Bubble* yang dicari tetangganya beserta tetangga yang ditemukan ditampung dalam suatu array. Apabila jumlah tetangga *bubble* M atau N yang ditemukan telah mencapai 3 atau lebih dari 3 *bubble*, maka *bubble* beserta tetangga yang ditemukan akan diledakkan.

3. Pencarian semua *bubble* yang menggantung agar dapat dijatuhkan dilakukan dengan cara melakukan pencarian tetangga terhadap semua *bubble* yang berada di arena permainan. Warna tidak menjadi acuan dalam pencarian tetangga ini. Definisi *bubble* menggantung dalam hal ini adalah *bubble* yang tidak berada pada baris pertama, atau *bubble* yang tidak memiliki tetangga yang berada pada baris pertama. Sama halnya dengan pencarian tetangga *bubble* untuk diledakkan, grup *bubble* yang ditemukan ditampung dalam suatu array. Apabila grup *bubble* yang ditemukan tidak ada yang berada pada baris pertama arena, maka grup *bubble* tersebut akan dijatuhkan.
4. Dalam *game Puzzle Bubble Hawaii* ini terdapat 3 tipe *game*, yaitu *arcade*, *time trial*, dan *battle*. Tingkat kesulitan yang diterapkan pada *game* ini untuk merintangai pemain dalam menyelesaikan permainan, yaitu *easy*, *normal*, dan *hard*. Tiap tingkat kesulitan terdiri atas 10 level (pada tipe *game arcade*) dan 1 level (pada tipe *game time trial* dan *battle*). Hal yang membedakan antar level dalam suatu tingkat kesulitan terletak pada banyaknya warna *bubble*, kecepatan turun *bubble* menurut waktu tertentu, ada atau tidaknya *unexploded bubble* dan *magic bubble*, jumlah skor yang harus dicapai untuk memenangkan suatu level, dan jumlah *bubble* yang harus diledakkan untuk memenangkan suatu level.
5. Fitur *multiplayer* yang diterapkan dalam aplikasi *game* ini dapat berjalan dengan baik. Pada permainan *multiplayer*, pemain dapat saling menyerang dengan mengirimkan *bubble* ke arena musuh setiap mencapai skor tertentu.

2 Saran

1. Pengecekan tabrakan antara *bubble* dengan dinding arena serta *bubble* pada arena pada *game Puzzle Bubble Hawaii* ini menggunakan metoda `hitTest()`. Namun pengecekan tabrakan dirasa masih belum akurat. Sehingga disarankan pada pengembangan selanjutnya, pengecekan tabrakan dapat diperbaiki agar pendeteksian objek lebih akurat dengan menggunakan metoda `getBounds()`.
2. Keefisienan dalam hal *coding* yang diterapkan pada *game Puzzle Bubble* ini dinilai masih kurang. Hal ini tampak pada performa *game* ketika *game* dijalankan dalam jangka waktu yang lama, akan terjadi *lag* (program berjalan lebih lambat dari yang

seharusnya). Oleh karena itu, *coding* dapat diperbaiki sehingga *game* ini baik dalam hal performa maupun penggunaan memori dapat menjadi lebih efektif.