

Penerapan Algoritma *Linear Congruent Method* Terhadap *Spawning Enemy*

Muttaqin¹, Atthariq², Mursyidah³

^{1,3} Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe
Jln. B.Aceh Medan Km.280 Buketrata 24301 INDONESIA

E-mail : taqin25071995@gmail.com

Abstrak—Keberadaan *NPC (Non Player Character)* dalam suatu game merupakan salah satu faktor penting dalam menentukan game itu menarik atau tidak. Jika tidak ada penciptaan *NPC* yang cerdas dimana *NPC* tidak bersifat adaptif, dan reaktif, maka jika dilihat dari sisi game dapat dikatakan tidak menarik dan cenderung membosankan, sehingga penulis bertujuan membuat penelitian yaitu perancangan dan pembuatan *Game Survival Shooter 3D* dengan studi kasus *Random* jumlah objek menggunakan algoritma *LCM (Linear Congruent Method)*. Kejadian acakan tersebut terjadi pada acakan kemunculan musuh setelah berburu. permainan ditugaskan untuk berfokus menghindari, dan menembak *NPC* untuk terus bertahan hidup selama mungkin dengan alat seadanya dan melanjutkan misi hingga sampai ke akhir permainan. sehingga menciptakan game *AI* tiga dimensi (3d) yang mampu menguji kecepatan, reaksi dari pemainnya, dan meningkatkan intelektual bagi penggunanya.

Kata kunci : *NPC (Non Player Character), Survival Shooter, 3D, LCM(Linear Congruent Method), Artificial Intelligence.*

Abstract—The existence of *NPC (Character Non Player)* in a game is one important factor in determining where game is interesting or not. If there is no intelligent *NPC* creation where *NPC* are not adaptive and reactive when it is viewed from the side of a gameplay, it can be said not interesting and tend to be boring. So the author aims to make research that designing and making *Game Survival Shooter 3D* with the case study *Random* number of objects using *LCM (Linear Congruent Method)* algorithm. The incident happened at the random appearance of the enemy after player has killed an enemy. Players are assigned to focus on avoiding and shooting *NPCs* to stay alive for as long as possible with makeshift tools are and continue the mission until the end of the game. Thereby creating a three-dimensional *AI* game (3d) that is able to test the speed, reaction of its players, and enhance the intellectual for its users.

Keyword : *NPC (Non Player Character), Survival Shooter, 3D, LCM(Linear Congruent Method), Artificial Intelligence.*

I. PENDAHULUAN

Game ataupun permainan merupakan sistem dimana pemain terlibat konflik buatan, pemain berinteraksi dengan sistem dan konflik dalam permainan merupakan rekayasa atau buatan [1].

Artificial Intelligence adalah bagaimana membuat komputer mampu berpikir dan melakukan kegiatan seperti halnya manusia atau binatang [2]. perkembangan kecerdasan buatan pada game komputer dan *videogame* semakin cepat dan memanjakan para pemainnya dengan teknologi terkini yang merupakan perkara menarik untuk dicermati, sehingga game bukan hanya sekedar hobi untuk mengisi waktu luang, melainkan sebuah cara untuk meningkatkan kreativitas dan tingkat intelektual penggunaannya [3].

Genre game adalah klasifikasi game yang didasari interaksi pemainnya. salah satu genre game yang diminati yaitu *Game Survival Shooter* tiga dimensi (3D) yang menguji kecepatan dan reaksi dari pemainnya, dengan *sub genre Third Person Shooter (TPS)* yang memberikan pandangan *perspektif* pihak ketiga atau karakter *player* dapat terlihat sepenuhnya sehingga memberikan pemain pandangan yang lebih luas akan lingkungan sekitar [4].

Biasanya tujuan dari permainan *Survival Game* yaitu menghindar dan menembak lawan untuk terus bertahan hidup selama mungkin dengan alat seadanya dan melanjutkan misi hingga sampai ke tujuan sebenarnya [5].

Game yang tidak menerapkan Artificial intelligence dalam permainan, user cenderung cepat bosan dikarenakan user dapat menebak bahkan telah mengetahui posisi dan kekuatan musuh yang dihadapinya. sehingga game kurang menarik untuk dimainkan.

Game kecerdasan buatan Artificial Intelligence mengacu pada teknik yang digunakan dalam permainan komputer dan video untuk menghasilkan ilustrasi intelijen terhadap Musuh yang diterapkan pada game saat ini. *spawning enemy* atau kemunculan musuh salah satu elemen game yang mengatasi pembagian monster dalam tiap area dalam peta. teknik *spawning* ini juga bermacam-macam, ada yang dilakukan secara acak atau secara teratur ketika dibangkitkat dan pemain dapat memburu atau menguji kemampuannya untuk melawan musuh sesuai kondisi yang dihadapi.

Dengan menggunakan teknik *linear congruential method* yang merupakan pembangkit bilangan acak yang sederhana, mudah dimengerti teorinya, dan juga mudah untuk diimplementasikan dimana memperoleh bilangan acak yang seakan-akan tidak terjadi pengulangan, dengan melakukan beberapa pengujian yang akan diterapkan kepada kemunculan jumlah monster pada area peta yang akan dimulai dengan batasan-batasan dan modifikasi yang ada. kemunculan jumlah musuh secara random akan di munculkan pada periode waktu tertentu ketika musuh sudah dikalahkan oleh pemain, maka jumlah random musuh akan dimunculkan dengan menyebar

sesuai periode waktu yang telah ditentukan sehingga area tetap terisi.

II. METODOLOGI PENELITIAN

A. Linear Congruent Method(LCM)

I Putu Gede Budayasa dan Gusde Paryatna (2013) *Linear Congruential Method* atau jika diubah kedalam bahasa Indonesia menjadi Pembangkit Bilangan Acak Kongruen-Lanjar merupakan pembangkit bilangan acak yang sederhana, mudah dimengerti teorinya, dan juga mudah untuk diimplementasikan^[6]. LCM didefinisikan dalam relasi berulang berikut :

$$X_n = (a * X_{n-1} + c) \text{ mod } m \dots \dots \dots (2.1)$$

Dimana :

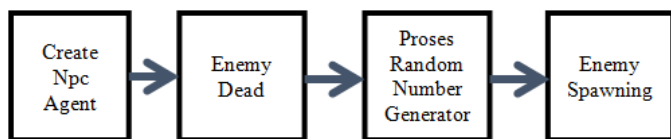
- X_n = bilangan acak ke-n dari deretnya
- X_{n+1} = bilangan acak sebelumnya ($0 \leq x_n \leq m$)
- a = faktor pengali ($0 \leq a \leq m$)
- c = increment ($0 \leq c \leq m$)
- m = modulus ($0 \leq m$)

Pada algoritma di atas, X_n merupakan variabel Bilangan acak ke-n, dimana a dan c sebagai konstanta LCM dan m sebagai batas maksimum bilangan acak. Salah satu sifat dari metode ini adalah terjadi pengulangan pada periode waktu tertentu atau setelah sekian kali pembangkitan.

Untuk mengatasi terjadinya pengulangan tersebut maka penentuan konstanta LCM (a , c dan m) sangat menentukan baik tidaknya bilangan acak yang diperoleh dalam arti memperoleh bilangan acak yang seakan-akan tidak terjadi pengulangan, dengan melakukan beberapa pengujian^[7].

B. Blok Diagram

Pada penelitian ini, akan dibahas proses pengacakan jumlah musuh pada game. jika musuh berhasil dikalahkan oleh pemain maka musuh akan dimunculkan kembali dengan jumlah acakan yang sudah dideklarasikan. seperti yang ditunjukkan pada gambar 2.1. berikut ini.



Gambar 2.1 Alur Proses Kemunculan Acak Jumlah Musuh

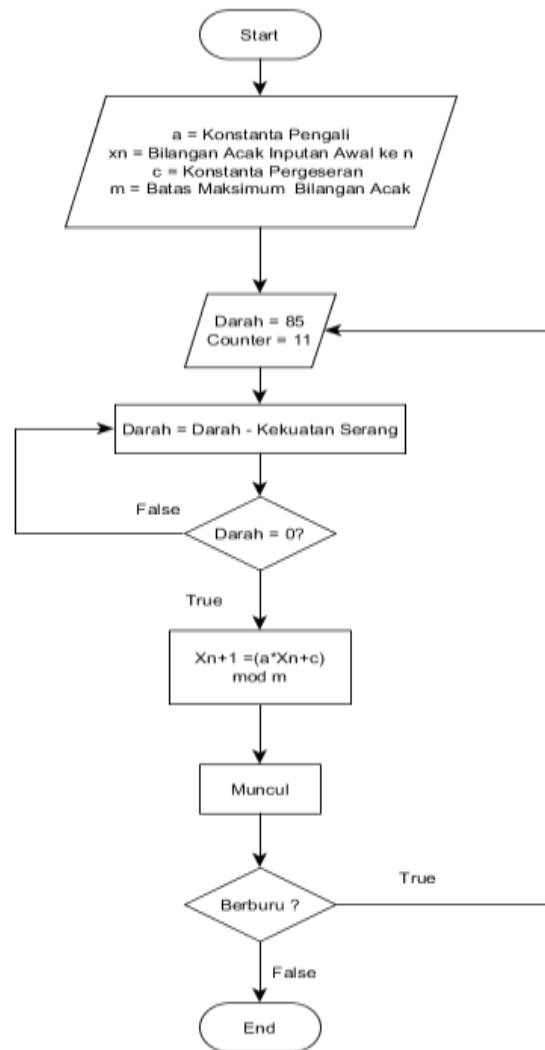
- 1) NPC
Proses ini yaitu muncul nya musuh AI atau *non-player character* pada awal mulai permainan.
- 2) Enemy Dead
Proses ini yaitu ketika Player memburu dan mengalahkan musuh sampai darah musuh bernilai 0.
- 3) Proses Random Number Generator
Proses ini akan mengambil nilai parameter status musuh untuk menentukan nilai acak menggunakan Linear Congruential Generator.
- 4) Enemy Spawning
Proses ini membangkitkan musuh dimana Jumlahnya sudah diacak sesuai deklarasi yang sudah ditentukan.

C. Flowchart

Pada bagian ini ditunjukkan proses kemunculan jumlah npc atau musuh secara acak, dapat dilihat pada gambar 2.2. sebagai berikut ini.

Gambar 2.2 Diagram Alir Kemunculan Acak Jumlah Musuh

Pada gambar 2.2. merupakan flowchart untuk proses kemunculan jumlah musuh yang akan dibangkitkan secara acak ketika player berhasil mengalahkan salah satu musuh



dalam permainan game 3d survival TPS.

- 1) Penentuan konstanta LCM.
Pada bagian ini, user melakukan inputan nilai untuk menentukan baik tidaknya bilangan acak yang diperoleh dalam arti memperoleh bilangan acak yang seakan-akan tidak terjadi pengulangan.
- 2) Status Musuh.
selanjutnya ada 2 variable yang akan dijadikan nilai konstanta dimana a sebagai darah musuh dan c sebagai nilai counter musuh.
- 3) Serangan Player.
Dalam proses ini, player mencari dan menyerang musuh dengan kekuatan tembak dimana Musuh memiliki Status Parameter Darah yang berbeda.
- 4) Darah Musuh.

Pada kondisi ini ketika nilai darah musuh mencapai 0 dikarenakan terkena tembakan oleh player maka musuh akan mati jika darah musuh lebih besar dari 0 maka player akan menyerangnya sampai musuh akan mati.

5) .Linear Congruential Method

Saat player sudah mengalahkan musuh, variable konstanta musuh akan diproseskan untuk menghitung nilai acak awal sampai nilai acak-n yang akan didapatkan.

6) .Kemunculan musuh

Nilai acak yang sudah didapatkan akan menentukan jumlah musuh yang akan dibangkitkan ke dalam arena sesuai deklarasi yang di tentukan.

7) Berburu musuh

Pada kondisi ini, jika player ingin masih berburu atau menguji kemampuan user maka player tetap harus mengalahkan salah satu musuh untuk membangkit musuh yang baru.

III. HASIL DAN PEMBAHASAN

A. Objek Model Zombear.

Pada table 3.1. merupakan rincian status npc zombear memiliki nilai darah 85, mempunyai nilai score 5 jika sudah berhasil dikalahkan dan nilai counter 11. Serangan pukulan Zombear bernilai 5 dan kecepatan berjalan 10.

Status Health	
Health monster	85
Score value	5
Counter value	11
Status Attack	
Attack damage	5
Movement	
Run speed	10

Tabel 3.1. Status Zombear Non-Player Character.

B. Hasil Pengujian Data Dan Analisis

Pada tahap pengujian data dimana yang dianalisis adalah untuk menentukan hasil acakan jumlah kemunculan musuh yang telah dirancang dengan bahasa pemograman C#, dimana bilangan yang kemunculannya terjadi secara acak, yang digunakan pada simulasi game yang terdapat pada Game Online ketika berburu musuh untuk menghasilkan sesuatu. Bilangan yang akan dibangkitkan menggunakan metode Linear Congruential method (LCM) salah satu sebagai metode untuk menentukan kemunculan acakan.

Saat player berhasil mengalahkan salah satu musuh dan mensinkronisasi nilai dari nilai counter musuh sebagai c dan mensinkronisasi nilai dari darah musuh sebagai a, dimana darah musuh dan nilai counter merupakan nilai konstanta.

Setelah sinkronisasi dinisialisasi, sehingga digunakanlah proses acakan sebagai berikut ini :

a) Pengujian Kemunculan Zombear Secara Acak

1. Ketika player mengalahkan Zombear
Maka, a=85, xn-1=0, c=11, m=7
Dimana :

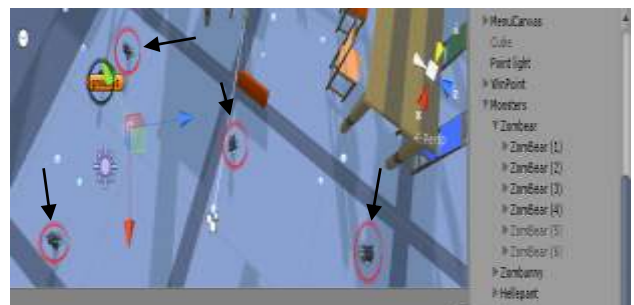
a adalah darah musuh yang dikorelasikan.
c adalah nilai counter yang dikorelasikan.
m adalah jumlah batas kemunculan musuh yang dibangkitkan.
xn adalah bilangan ke-n dari deretnya.
xn-1 adalah bilangan acak sebelumnya.

proses:

$$X_n = (85 \cdot 0 + 11) \text{ mod } 7$$

$$X_n = 11 \text{ mod } 7$$

$$X_n = 4$$



Gambar 3.2. Kemunculan Jumlah Musuh pada Acakan Pertama

Pembuktian bawah 4 Zombear telah dimunculkan di peta yang ditandai dengan anak panah dan diaktifkan pada Hierarchy window pada acakan pertama pada gambar 3.2 diatas.

2. Ketika player mengalahkan Zombear
Maka a=85, xn-1=4, c=11, m=7

Proses :

$$X_n = (85 \cdot 4 + 11) \text{ mod } 7$$

$$X_n = 351 \text{ mod } 7$$

$$X_n = 1$$



Gambar 3.3. Kemunculan Jumlah Musuh pada Acakan Kedua

Pembuktian bawah 1 Zombear telah dimunculkan di peta yang ditandai dengan anak panah dan diaktifkan pada Hierarchy window pada acakan kedua pada gambar 3.3 diatas.

3. Ketika player mengalahkan Zombear
Maka a=85, xn-1=1, c=11, m=7

Proses:

$$X_n = (85 \cdot 1 + 11) \bmod 7$$

$$X_n = 96 \bmod 7$$

$$X_n = 5$$



Gambar 3.4. Kemunculan Jumlah Musuh pada Acakan Ketiga

Pembuktian bawah 5 Zombear telah dimunculkan di peta yang ditandai dengan anak panah dan diaktifkan pada Hierarchy window pada acakan ketiga pada gambar 3.4 diatas.

4. Ketika player mengalahkan Zombear
Maka $a=85$, $x_n-1=5$, $c=11$, $m=7$

Proses :

$$X_n = (85 \cdot 5 + 11) \bmod 7$$

$$X_n = 436 \bmod 7$$

$$X_n = 2$$



Gambar 3.5. Kemunculan Jumlah Musuh pada Acakan Keempat

Pembuktian bawah 2 Zombear telah dimunculkan di peta ditandai dengan anak panah dan diaktifkan pada Hierarchy window pada acakan keempat pada gambar 3.5 diatas.

5. Ketika player mengalahkan Zombear
Maka $a=85$, $x_n-1=2$, $c=11$, $m=7$

Proses:

$$X_n = (85 \cdot 2 + 11) \bmod 7$$

$$X_n = 181 \bmod 7$$

$$X_n = 6$$



Gambar 3.6. Kemunculan Jumlah Musuh pada Acakan Kelima

Pembuktian bawah 6 Zombear telah dimunculkan di peta yang ditandai dengan anak panah dan diaktifkan pada Hierarchy window pada acakan kelima pada gambar 3.6 diatas.

6. Ketika player mengalahkan Zombear
Maka $a=85$, $x_n-1=6$, $c=11$, $m=7$

Proses:

$$X_n = (85 \cdot 6 + 11) \bmod 7$$

$$X_n = 521 \bmod 7$$

$$X_n = 3$$



Gambar 3.7. Kemunculan Jumlah Musuh pada Acakan Keenam

Pembuktian bawah 3 Zombear telah dimunculkan di peta yang ditandai dengan anak panah dan diaktifkan pada Hierarchy window pada acakan keenam pada gambar 3.7 diatas.

7. Ketika player mengalahkan Zombear
Maka $a=85$, $x_n-1=3$, $c=11$, $m=7$

Proses:

$$X_n = (85 \cdot 3 + 11) \bmod 7$$

$$X_n = 266 \bmod 7$$

$$X_n = 0$$



Gambar 3.7. Kemunculan Jumlah Musuh pada Acakan Ketujuh

Pembuktian bawah zombear tidak dibangkitkan sedikitpun di peta dan terlihat nonactive zombear pada Hierarchy window pada acakan ketujuh pada gambar 3.7 diatas.

Tabel 3.2. Hasil Jumlah Kemunculan Musuh Zombear..

Proses	Hasil Kemunculan Musuh
jika $a=85$, $xn-1=0$, $c=11$, dan $m=7$	4 Zombear dibangkitkan
jika $a=85$, $xn-1=4$, $c=11$, dan $m=7$	1 Zombear dibangkitkan
jika $a=85$, $xn-1=1$, $c=11$, dan $m=7$	5 Zombear dibangkitkan
jika $a=85$, $xn-1=5$, $c=11$, dan $m=7$	2 Zombear dibangkitkan
jika $a=85$, $xn-1=5$, $c=11$, dan $m=7$	6 Zombear dibangkitkan
jika $a=85$, $xn-1=6$, $c=11$, dan $m=7$	3 Zombear dibangkitkan
jika $a=85$, $xn-1=3$, $c=11$, dan $m=7$	0 Zombear dibangkitkan

Pada tabel 3.2. merupakan hasil acakan awal sampai akhir dalam sekali pembangkitan. pada acakan pertama terjadi 4 kemunculan zombear, acakan kedua membangkitkan 1 zombear, acakan ketiga membangkitkan 5 zombear, acakan keempat membangkitkan 2 zombear, acakan kelima membangkitkan 6 zombear, acakan keenam membangkitkan 3 zombear dan acakan ketujuh tidak membangkitkan apapun.

IV. KESIMPULAN

Berdasarkan perancangan game (3d) *Third Person Shooter Survive Shooter* mengenai simpulan yang di hasilkan dari pengujian pada Penerapan Algoritma Linear Congruent Method Terhadap Spawning Enemy, dapat disimpulkan bahwa Terjadi kemunculan pada setiap zombear terjadi pengulangan pada periode waktu tertentu atau setelah sekian kali pembangkitan dan Penentuan konstanta LCM (a , c , dan m) sangat menentukan baik tidaknya bilangan acak yang diperoleh dalam arti memperoleh bilangan acak yang seakan-akan tidak terjadi pengulangan.

REFERENSI

- [1] Lestari, D. 2012. Game Advanture Misteri Kotak Pandora. (Online) <http://ejournal.amikomputerwokerto.ac.id/index.php/telematika/article/download/247/222>
- [2] Millington Dan Funge. 2009. *Artificial intelligence for games*. (Online) http://lecturer.ukdw.ac.id/~mahas/dossier/gameng_AIFG.pdf
- [3] Edi Wijaya. 2013. Analisis Penggunaan Algoritma Breadth First Search Dalam Konsep Artificial Intellegence. (Online) <http://ejournal.stmik-time.ac.id/index.php/jurnalTIMES/article/download/6/4>
- [4] Ivan C. Sibero. 2009. Langkah - langkah Mudah Membuat Game 3D. Yogyakarta Penerbit MediaKom
- [5] Kleinbaum. 1996. Survival analysis : a self learning text. (Online) <http://download.portalgaruda.org/article.php?article=128536&val=1185>. Diakses Agustus 2017

- [6] I Putu Gede Budayasa dan Gusde Paryatna. 2013. Implementasi Linear Congruent Generator (LCG) Dalam Rancang Bangun Aplikasi Game Peduli Lingkungan. (Online) <https://ejournal.undiksha.ac.id/index.php/janapati/article/view/9765>

- [7] Dian Sekarsari. 2014. Metode LCM (Linear Congruent Method) Pada Permainan Ludo. (Tugas Akhir Mahasiswa Program Studi Teknik Informatika STMIK Budi Darma Medan) (Online) <http://megaslides.top/doc/347608/implementasi-metode-lcm--linear-congruent-method--pada>.

