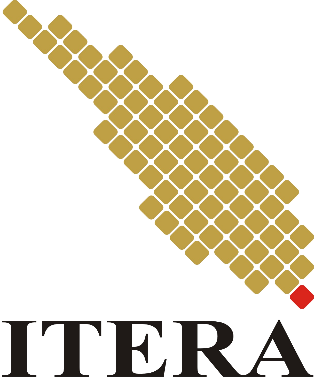
****

LAPORAN TUGAS BESAR

ALGORITMA DAN STRUKTUR DATA

Disusun Oleh :

NURTIAS RAHAYU (14117086)

ROBBY LEGITRA KURNIAWAN (14117101)

ARIMBI AYUNINGTYAS (14117140)

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN SAINS

INSTITUT TEKNOLGI SUMATERA

LAMPUNG SELATAN

2018

# **HALAMAN PENGESAHAN**

**LAPORAN KEGIATAN**

**MEMBUAT GAME HANGMAN**

**DENGAN MENGGUNAKAN PEMROGRAMAN C++**

**TANGGAL 28 NOVEMBER S/D 16 DESEMBER 2018**

Yang bertanda tangan di bawah ini :

Nama Asisten : Syahid Prabowo

NIM : 14116010

Menyatakan nama-nama di bawah ini :

Kelompok 4

Ketua Kelompok : Nurtias Rahayu

NIM : 14117086

Anggota :

1. Robby Legitra Kurniawan (14117101)
2. Arimbi Ayuningtyas (14117140)

Bahwa benar laporan dan program yang dibuat merupakan karya sendiri dan *programming project* yang telah di kumpulkan akan menjadi hak milik Program Studi Teknik Informatika.

Lampung Selatan, 14 Desember 2018

Mengetahui

Ketua Kelompok Asisten

Nurtias Rahayu Syahid Prabowo  
 (14117086) (14116010)

# **KATA PENGANTAR**

Puji syukur atas kehadirat Allah SWT yang telah memberikan rahmat dan karunia-Nya sehingga penyusun dapat menyelesaikan Penulisan Laporan Tugas Besar dengan tema “GAME HANGMAN” dengan tepat waktu. Diharapkan dengan dibuatnya laporan ini khhususnya para pembaca dapat menambah wawasan dan ilmu pengetahuan mengenai pemrograman komputer.

Kami mengucapkan terima kasih kepada Bpk. Ahmad Luky Ramdani,S.Kom.,M.Kom. dan Bpk. Arief Ichwani,S.Kom.,M.Cs. selaku dosen pengampu matakuliah Algoritma dan Struktur Data, Bpk. I Wayan Wiprayoga W,S.Kom.,M.Kom. selaku dosen pengampu *project game hangman*, Syahid Prabowo selaku asisten pembimbing, serta rekan – rekan yang telah membantu dalam pembuatan *programming project* dan penyusunan laporan Tugas Besar.

Mengingat bahwa laporan ini masih jauh dari kata sempurna, oleh karena itu penyusun mengharapkan segala bentuk kritikan dan saran yang membangun dari pembaca guna meningkatkan kualitas dari karya tulis ini.

Penyusun juga mengucapkan terima kasih kepada seluruh pihak yang telah berperan dalam penyusunan laporan ini. Semoga Allah senantiasa meridhoi usaha yang telah dilakukan.

Lampung Selatan,28 November 2018

Penyusun

# **TAKARIR**

*Game* = Permainan

*Source Code =* Kode Sumber

*User*  = Pengguna

*Programming Project* = Pembuatan Program

*Random* = Acak

*Artificial Intelegence* = Kecerdasan Buatan

*Software* = Perangkat Lunak

*Linked List*  = Daftar Berantai

*Flowchart* = Diagram Alir

*Capture* = Menangkap

*Hint*  = Petunjuk

**DAFTAR ISI**

[**HALAMAN PENGESAHAN** ii](#_Toc532756130)

[**KATA PENGANTAR** iii](#_Toc532756131)

[**TAKARIR** iv](#_Toc532756132)

[**DAFTAR GAMBAR** vi](#_Toc532756133)

[**BAB I PENDAHULUAN** 1](#_Toc532756134)

[1.1 Latar Belakang 1](#_Toc532756135)

[1.2 Batasan Masalah 1](#_Toc532756136)

[1.3 Manfaat Program 1](#_Toc532756137)

[1.4 Tujuan Penelitian 2](#_Toc532756138)

[1.5 Sistematika Penulisan 2](#_Toc532756139)

[**BAB IIMETODOLOGI** 3](#_Toc532756140)

[2.1 Analisis Program 3](#_Toc532756141)

[2.2 Penulisan Algoritma 4](#_Toc532756142)

[**BAB III HASIL DAN PEMBAHASAN** 6](#_Toc532756143)

[3.1 Penjelasan Program Berserta *Capture* Hasil Program 6](#_Toc532756144)

[3.2 Kekurangan Dan Kelebihan Program 9](#_Toc532756145)

[3.3 *Source Code* 9](#_Toc532756146)

[**BAB IV KESIMPULAN DAN SARAN** 20](#_Toc532756147)

[4.1 Kesimpulan 20](#_Toc532756148)

[4.2 Saran 20](#_Toc532756149)

[**DAFTAR PUSTAKA** 21](#_Toc532756150)

[**LAMPIRAN** 22](#_Toc532756151)

# **DAFTAR GAMBAR**

[Gambar 3. 1 Tampilan Loading 6](#_Toc532598021)

[Gambar 3. 2 Tampilan Menu 6](#_Toc532598022)

[Gambar 3. 3 Petunjuk Permainan 7](#_Toc532598023)

[Gambar 3. 4 Pemain memilih lawan bermain 7](#_Toc532598024)

[Gambar 3. 5 User menebak kata yang terdapat pada kamus 7](#_Toc532598025)

[Gambar 3. 6 Tampilan saat user menebak dengan salah 8](#_Toc532598026)

[Gambar 3. 7 Tampilan user pemain menebak dengan benar 8](#_Toc532598027)

[Gambar 3. 8 Tampilan saat pemain memasukkan nama dan kata yang ditebak 8](#_Toc532598028)

[Gambar 3. 9 Tampilan saat pemain kedua menebak kata 9](#_Toc532598029)

# **BAB I PENDAHULUAN**

## Latar Belakang

Bahasa Inggris adalah bahas universal yang digunakan hampir di seluruh dunia. Di Negara kita pun sudah banyak yang mengerti dan memahami bahasa tersebut. Banyak cara-cara untuk mempelajari bahasa Inggris, seperti mengambil kursus, membaca kamus, belajar bahasa Inggris *online*, maupun memainkan *game* yang berhubungan dengan bahasa Inggris. Saat ini sudah sangat banyak bermunculan *game* berbasis bahasa Inggris yang ada, salah satunya adalah *game hangman* atau tebak kata dalam bahasa Inggris.

Banyak orang yang memandang sebelah mata mengenai *game* yang ada sekarang. Banyak sisi negatif yang bermunculan karena kecanduan bermain *game*. Namun, adapula sisi positif yang dapat diambil dengan adanya *game*. Salah satunya sebagai media pembelajaran. Dengan *game* yang memiliki daya tarik sendiri itu artinya pembelajaran bisa jadi lebih efektif. Dengan permainan tebak kata ini, *user* bisa menambah kosakatanya dalam bahasa inggris.

## Batasan Masalah

Pada penelitian ini perlu didefinisikan beberapa batasan sejauh mana penelitian ini akan dikerjakan. Batasan-batasan masalah tersebut adalah sebagai berikut:

1. Implementasi permainan Hangman ini dirancang menggunakan bahasa pemrograman C++.
2. Kata yang ditebak hanya satu macam, contohnya : nama hewan.
3. Kata yang ditebak dalam bahasa Inggris.
4. Jumlah huruf dalam sebuah kata yaitu minimal enam huruf dan maksimal empat belas huruf.
5. Permainan ini diperuntukan untuk siapa saja.

## Manfaat Program

Manfaat dalam pembuatan program adalah

Bagi pembuat program :

Dapat memahami dan mengimplementasikan *linked* *list* sebagai sarana pemecahan masalah di suatu program.

Bagi *User* :

Dapat menambah kosakata dalam bahasa inggris.

## Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Mengimplementasikan algoritma Linked List sebagai kecerdasan buatan pada permainan Hangman.
2. Merancang permainan yang mengandung unsur pendidikan.
3. Menambah kosakata *user* dalam bahasa Inggris.

## Sistematika Penulisan

Untuk mempermudah penyusunan dan analisa program tugas besar ini, maka penulis membuat sistematika laporan yang menjadi penyusunan dalam menyelesaikan tugas besar ini :

**BAB I PENDAHULUAN**  
Bab ini menjelaskan tentang latar belakang,batasan masalah,manfaat program,tujuan program, dan sistematika penulisan program.

**BAB II METODOLOGI**  
Bab ini berisi tentang analisa program dan penulisan algoritma program.

**BAB III HASIL DAN PEMBAHASAN**  
Bab ini berisi tentang penjelasan program,kekurangan dan kelebihan program,serta source code aplikasi Game Hangman.

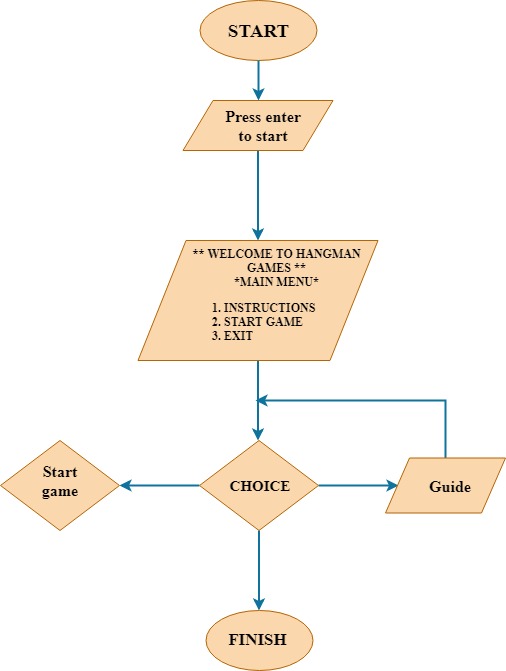
**BAB IV KESIMPULAN DAN SARAN**  
Bab ini berisi kesimpulan dan saran dari dibuatnya program aplikasi Game Hangman.

# **BAB IIMETODOLOGI**

## 2.1 Analisis Program

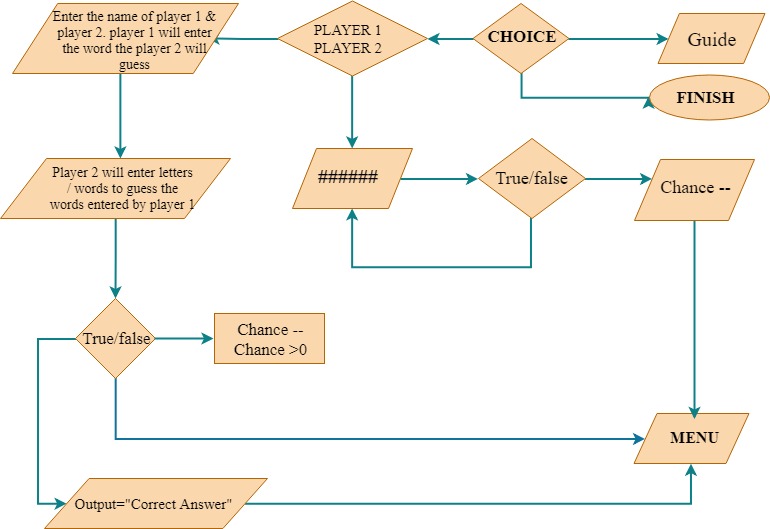
Rancangan awal dari program aplikasi *Game* Hangman ini adalah sebuah program yang memiliki fitur bermain dengan kata yang sudah disediakan oleh komputer, bermain dengan teman, dan dapat melihat skor yang telah diperoleh. *Game* ini bertujuan untuk menebak sebuah kata, baik kata yang sudah ada dalam kamus maupun yang baru saja di inputkan oleh lawan main, pemain di berikan 4 kali kesempatan untuk menjawab. Sebelum memulai permainan, awal tampilan menu terdapat tiga pilihan, yaitu *“How to Play Game Hangman?”*, *“Play Game”*, dan *“Quit Game”*. Jika kita memilih *“How to Play Game Hangman”*, maka akan menampilkan penjelasan tentang cara bermain permainan ini, sedangkan saat kita memilih *“Play Game”*, kita bisa memilih apakah kita akan bermain bersama komputer atau dengan teman. Jika *user* memilih bermain bersama teman, maka pemain pertama akan memasukan kata yang akan ditebak oleh pemain kedua, sedangkan jika *user* memilih bermain bersama komputer, maka pemain akan menebak kata yang sudah tersedia di dalam kamus secara *random*.

## 2.2 Penulisan Algoritma

****

**Flowchart Menu Utama**

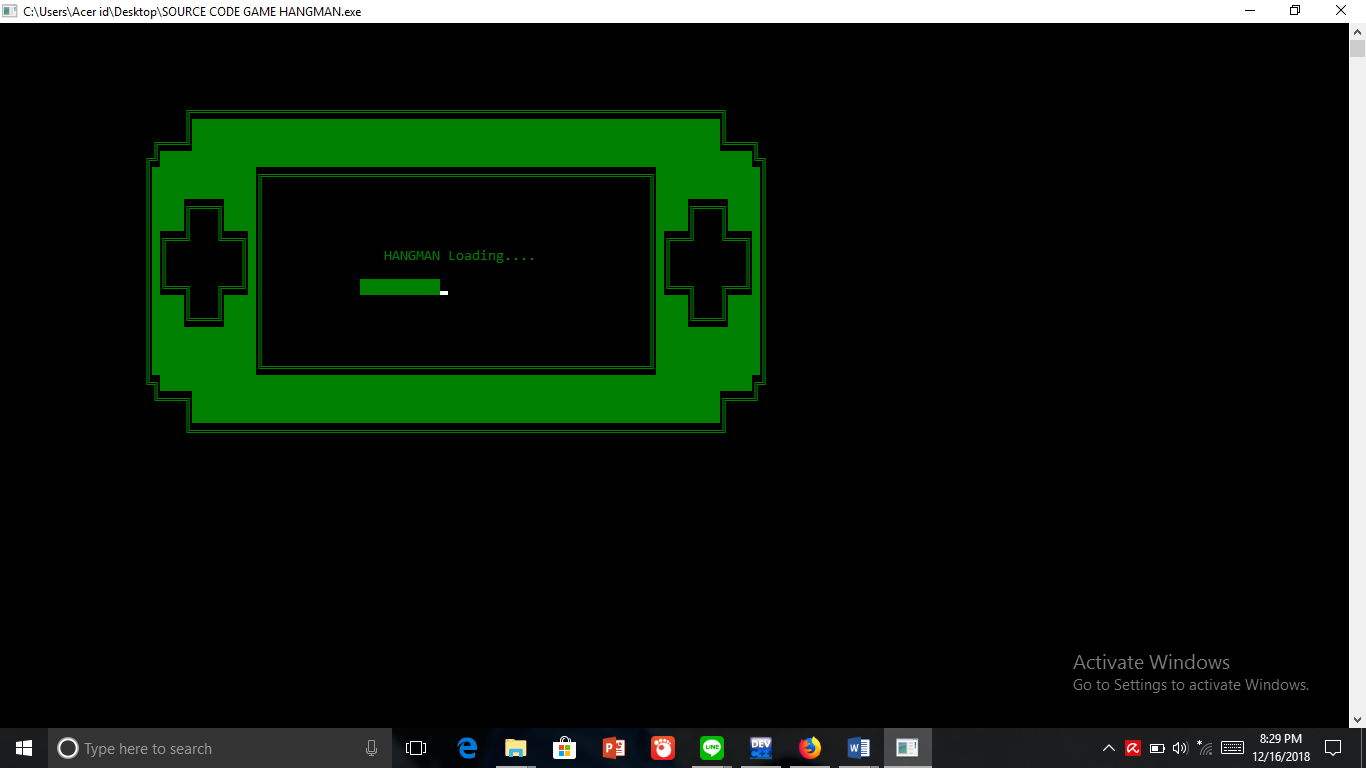
**Game Hangman**

****

# **BAB III HASIL DAN PEMBAHASAN**

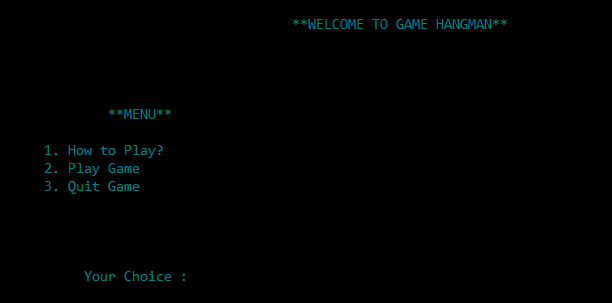
## 3.1 Penjelasan Program Berserta *Capture* Hasil Program

1. Program dimulai dengan tampilanLoading



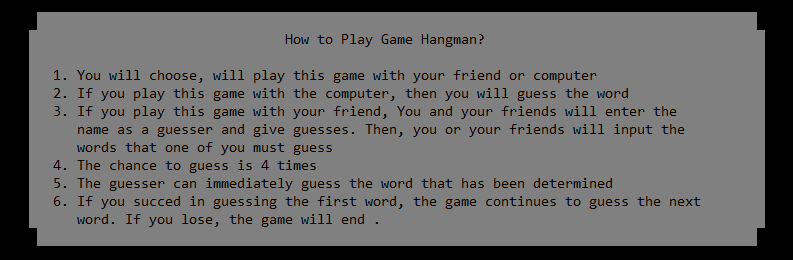
**Gambar 3. 1 Tampilan Loading**

1. Selanjutnya tampilan menu



**Gambar 3. 2 Tampilan Menu**

1. Petunjuk permainan Hangman, untuk memudahkan pemain memahami game ini.

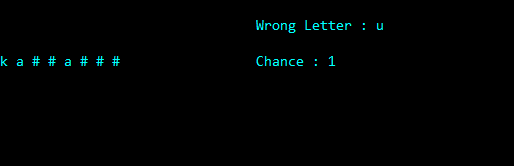


**Gambar 3. 3 Petunjuk Permainan**

1. Mulai bermain, disini *user* akan memilih akan bermain dengan komputer atau dengan teman.



**Gambar 3. 4 Pemain memilih lawan bermain**

1. Jika memilih bermain dengan komputer, maka *game* akan menampilkan kata yang sudah terdapat pada kamus.   
   

**Gambar 3. 5 User menebak kata yang terdapat pada kamus**

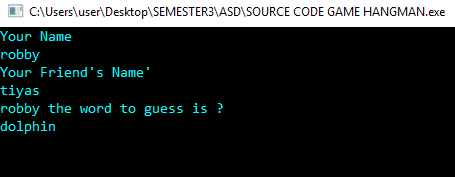
1. Jika *user* telah menggunakan empat kesempatannya dalam menebak kata maka akan muncul tampilan seperti ini



**Gambar 3. 6 Tampilan saat user menebak dengan salah**

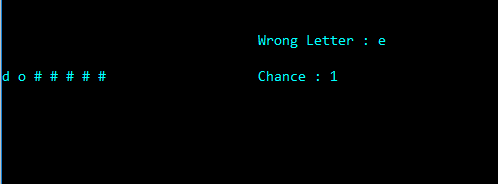


**Gambar 3. 7 Tampilan user pemain menebak dengan benar**

1. Saat pemain memilih bermain bersama teman, maka pemain harus memasukkan nama untuk pemain satu dan dua. Selanjutnya pemain pertama akan memasukkan kata untuk ditebak oleh pemain lainnya  
   

**Gambar 3. 8 Tampilan saat pemain memasukkan nama dan kata yang ditebak**

1. Pemain kedua akan menebak kata yang telah diinputkan oleh pemain pertama



**Gambar 3. 9 Tampilan saat pemain kedua menebak kata**

## 3.2 Kekurangan Dan Kelebihan Program

1. **Kekurangan Program**
2. Tidak terdapat *hint* atau bantuan.
3. Kosa kata dalam kamus terlalu sedikit.
4. **Kelebihan Program**
5. Apabila menebak huruf yang salah maka huruf tersebut akan di tampilkan di layar.
6. Bisa di mainkan dengan dua pemain.
7. Pemain bisa melihat skor yang telah diperolehnya.

## 3.3 *Source Code*

Berikut adalah *source code* dari GAME HANGMAN.

1. #include <iostream>// for input and output
2. #include <conio.h>//for getch and system cls
3. #include <windows.h>//for colour
4. #include <cstdlib>//buat list
5. #include <ctime>//buat mengakses waktu
6. #include <string.h>//buat sensitive case
7. #include <unistd.h>// for usleep function
8. **using** **namespace** std;
10. **typedef** **char** infotype;
11. **typedef** **struct** TElmt \*address;
12. **typedef** **struct** TElmt{
13. infotype info;
14. address next;
15. } Elmt;
17. **struct** List{
18. address first;
19. };
21. #define First(L) (L).first
22. #define Next(P) (P)->next
23. #define Info(P)(P)->info
24. #define Nil NULL
26. **void** CreateEmpty(List \*L){
27. (\*L).first=NULL;
28. }
30. **bool** IsEmpty(List L){
31. **return** ((L).first==NULL);
32. }
34. address Allocation(infotype x){
35. address NewElmt;
36. NewElmt = (Elmt\*) malloc(**sizeof**(Elmt));
37. NewElmt->info = x;
38. NewElmt->next = NULL;
40. **return** NewElmt;
41. }
43. **void** Deallocation(address hapus){
44. free(hapus);
45. }
47. **void** InsertFirst(List \*L, infotype x){
48. address NewElmt;
49. NewElmt = Allocation(x);
51. **if**(NewElmt != NULL){
52. **if**(IsEmpty(\*L)){
53. (\*L).first = NewElmt;
54. }
55. **else**{
56. NewElmt->next = (\*L).first;
57. (\*L).first = NewElmt;
58. }
59. }
60. }
62. **void** InsertAfter(address \*PredElmt, infotype x){
63. address NewElmt;
64. NewElmt = Allocation(x);
66. **if**(NewElmt != NULL){
67. NewElmt->next = (\*PredElmt)->next;
68. (\*PredElmt)->next=NewElmt;
69. }
70. }
72. **void** InsertLast(List \*L, infotype x){
73. **if** (IsEmpty(\*L)) InsertFirst(&(\*L), x);
74. **else** {
75. address temp;
76. temp = (\*L).first;
78. **while**(temp->next!=NULL) temp = temp->next;
80. InsertAfter(&temp, x);
81. }
82. }
84. **void** Output(List L){
85. **if**(!IsEmpty(L)){
86. address temp =First(L);
87. **while**(Next(temp)!=NULL){
88. cout<<Info(temp) <<" ";
89. temp = Next(temp);
90. }
91. cout<<Info(temp);
92. }
93. }
95. **void** cekhuruf(List \*L1, List \*L2, string x, **int** \*m){
96. **char** a=x[0];
97. address P1=First(\*L1), P2=First(\*L2);
98. **while**(Next(P1)!=NULL){
99. **if**(Info(P1)==a){
100. Info(P2)=a;
101. \*m=1;
102. }
103. P1 = Next(P1);
104. P2 = Next(P2);
105. }
106. **if**(Info(P1)==a){
107. Info(P2)=a;
108. \*m=1;
109. }
110. }

113. **void** cekkata(List L1, List \*L2, string x, **int** \*z){
114. address P1=First(L1), P2=First(\*L2);
115. **int** i=0,j=0;
116. \*z=1;
117. **while**(i<x.length()){
118. **if**(Info(P1)==x[i]){
119. j++;
120. }
121. P1=Next(P1);
122. i++;
123. }
124. **if** (j!=i){
125. \*z=0;
126. }**else**{
127. \*z=1;
128. }
129. }
131. **const** **char** rocket[] =
132. "           \_\n\
133. /^\\\n\
134. |-|\n\
135. |H|\n\
136. |A|\n\
137. |N|\n\
138. |G|\n\
139. |M|\n\
140. /|A|\\\n\
141. / |N| \\\n\
142. |  |\*|  |\n\
143. `-\"\"\"-`\n\
144. ";
146. **int** setcolor (unsigned **short** color){
147. **HANDLE** hCon = GetStdHandle(STD\_OUTPUT\_HANDLE);
148. SetConsoleTextAttribute(hCon,color);
149. }
151. **int** gotoxy (**int** x, **int** y){
152. COORD coord;
153. coord.X = x;
154. coord.Y = y;
155. SetConsoleCursorPosition(GetStdHandle(STD\_OUTPUT\_HANDLE),coord);
156. }
158. **void** convert (string &s) {
159. **for** (**int** i=0; i<s.length(); i++){
160. s[i] = tolower(s[i]);
161. }
162. }
164. **void** petunjuk (){
165. system("cls");
166. setcolor (128);
168. gotoxy (14,4); cout << "                   ";
169. gotoxy (13,5); cout << "How to Play Game Hangman? ";
170. gotoxy (13,6); cout << "                    ";
171. gotoxy (13,7); cout << "   1. You will choose, will play this  game with your friend or computer  ";
172. gotoxy (13,8); cout << "   2. If you play this game with the  computer, then you will guess the word   ";
173. gotoxy (13,9); cout << "  chosen by the program    ";
174. gotoxy (13,9); cout << "   3. If you play this game with your  friend, You and your friends will enter the    ";
175. gotoxy (13,10);cout << "      name as a guesser and give
176. guesses. Then, you or your friends will input the ";
177. gotoxy (13,11);cout << " words that one of you must guess";
178. gotoxy (13,12);cout << " 4. The chance to guess is 4 times";
179. gotoxy (13,13);cout << "   5. The guesser can immediately
180. guess the word that has been determined  ";
181. gotoxy (13,14);cout << "   6. If you succed in guessing the
182. first word, the game continues to guess the next ";
183. gotoxy (13,15);cout << "      word. If you lose, the game
184. will end ";
185. gotoxy (14,16);cout << "             ";
186. }
188. **int** skor=0;
189. **void** single(){
190. List L1,L2,L3;
191. srand((unsigned)time(0));
192. **int** x=rand();
193. string A[10]={"crocodile", "rabbit", "buffalo", "giraffe",
194. "butterfly","turtle", "kangaroo", "elephant", "dragonfly", "ladybird" };
195. string hangman,masuk;
197. system("cls");
198. setcolor(11);
200. CreateEmpty(&L1);
201. CreateEmpty(&L2);
202. CreateEmpty(&L3);
203. hangman = A[x%9];
204. **int** z=0, kesempatan=4;
206. **for**(**int** i=0; i<hangman.length(); i++){
207. gotoxy(20,8); InsertLast(&L1,hangman[i]);
208. }
209. **for**(**int** i=0; i<hangman.length(); i++){
210. gotoxy(20,8); InsertLast(&L2,'#');
211. }
213. **while**(kesempatan>0){
214. gotoxy(10,7); cout<<"\t\t\tWrong Letter : ";Output(L3);
215. cout<<endl<<endl;
217. Output(L2);cout<<"\t\t\tChance : "<<kesempatan<<endl;
218. kesempatan--;
219. **int** m=0;
221. cin>>masuk;
222. convert(masuk);
224. **if**(masuk.length()==1){
225. cekhuruf(&L1,&L2,masuk,&m);
226. }**else** **if**(masuk.length()==hangman.length()){
227. cekkata(L1,&L2,masuk,&z);
228. }
230. system("cls");
231. **if**(m==0){
232. InsertLast(&L3,masuk[0]);
233. }
235. **if**(z==1){
236. **break**;
237. getch();
238. system ("cls");
239. }
241. }
242. cout<<endl;
244. setcolor(7);
245. gotoxy(10,5); cout << "                   " ;
246. Output(L2); cout << "                      ";
247. gotoxy(10,6); cout << "   The word is : ";
248. Output(L1);
250. **if**(z==1){
251. string pilihan;
252. skor = skor+100;
253. setcolor(180);
255. gotoxy(15,8); cout << "       CONGRATULATION!!!       ";
256. gotoxy(15,9); cout << "                 YOU WIN       ";
257. gotoxy(15,10);cout << "                               ";
258. gotoxy(15,11);cout << "     YOUR SCORE IS "<< skor << "         ";
260. gotoxy(15,20);cout << "98(Next)\t99(Quit)";
261. gotoxy(15,21);cin  >> pilihan;
263. **if**(pilihan=="98"){
264. setcolor(10);
265. system ("cls");
266. single();
268. }**else**{
269. setcolor(10);
270. system ("cls");
271. }
272. }**else**{
273. setcolor(71);
275. gotoxy(15,8); cout << "                               ";
276. gotoxy(15,9); cout << "          GAME OVER            ";
277. gotoxy(15,10);cout << "                               ";
278. gotoxy(15,11);cout << "       YOUR SCORE IS "<< skor << "       ";
280. getch();
281. }
282. setcolor(10);
283. }
285. **void** dobel(){
287. List L1,L2,L3;
289. CreateEmpty(&L1);
290. CreateEmpty(&L2);
291. CreateEmpty(&L3);
293. string main1, main2, kata, jawab;
294. **int** z=0, kesempatan=4;
296. system("cls");
297. setcolor (11);
299. cout << "Your Name " << endl;
300. cin >> main1;
301. cout << "Your Friend's Name' "<< endl;
302. cin >> main2;
304. cout << main1 << " the word to guess is ? "<< endl;
305. cin >> kata;
307. **for**(**int** i=0; i<kata.length(); i++){
308. InsertLast(&L1,kata[i]);
309. }
310. **for**(**int** i=0; i<kata.length(); i++){
311. InsertLast(&L2,'#');
312. }
314. **for** (**int** i=0; i<30; i++){
315. cout << endl;
316. }
318. **while**(kesempatan>0){
319. gotoxy(10,7); cout<<"\t\t\tWrong Letter : ";Output(L3);
320. cout<<endl<<endl;
322. Output(L2);cout<<"\t\t\tChance : "<<kesempatan<<endl;
323. kesempatan--;
324. **int** m=0;
326. cin>>jawab;
327. convert(jawab);
329. **if**(jawab.length()==1){
330. cekhuruf(&L1,&L2,jawab,&m);
331. }**else** **if**(jawab.length()==kata.length()){
332. cekkata(L1,&L2,jawab,&z);
333. }
335. system("cls");
336. **if**(m==0){
337. InsertLast(&L3,jawab[0]);
338. }
340. **if**(z==1){
341. **break**;
342. getch();
343. system ("cls");
344. }
346. }
347. cout<<endl;
349. setcolor(7);
351. gotoxy(10,5); cout << "                   " ;
352. Output(L2); cout << "                      ";
353. gotoxy(10,6); cout << "   The word is : ";
354. Output(L1);
356. **if**(z==1){
357. setcolor(180);
359. gotoxy(15,8); cout << "      CONGRATULATION!!!        ";
360. gotoxy(15,9); cout << "                YOU WIN        ";
361. gotoxy(15,10);cout << "                               ";
363. setcolor(10);
364. getch();
365. system ("cls");
366. }**else**{
367. setcolor(71);
369. gotoxy(15,8); cout << "                               ";
370. gotoxy(15,9); cout << "          GAME OVER            ";
371. gotoxy(15,10);cout << "                               ";
373. getch();
374. }
375. setcolor(10);
376. }
378. **void** roket(){
379. **for** (**int** i = 0; i < 50; i ++) printf("\n"); // jump to bottom of  console
380. printf("%s", rocket);
381. **int** j = 300000;
382. **for** (**int** i = 0; i < 50; i ++) {
383. usleep(j); // move faster and faster,
384. j = (**int**)(j \* 0.9); // so sleep less each time
385. printf("\n"); // move rocket a line upward
386. }
387. }


391. **void** menu(){
392. string pilihan, pemain;
393. system ("cls"); setcolor (3);
395. cout << endl << endl;
396. cout << "          " << endl;
397. cout << "    \*\*WELCOME TO GAME HANGMAN\*\*           " << endl;
398. cout << "          " << endl;
399. gotoxy(11,7); cout << "                          ";
400. gotoxy(11,8); cout << "         \*\*MENU\*\*         ";
401. gotoxy(11,9); cout << "                          ";
402. gotoxy(11,10);cout << " 1. How to Play?          ";
403. gotoxy(11,11);cout << " 2. Play Game             ";
404. gotoxy(11,12);cout << " 3. Quit Game             ";
405. gotoxy(11,13);cout << "                          ";
406. gotoxy(13,17); cout << "    Your Choice :   ";
408. cin >> pilihan;
410. **if**(pilihan=="1"){
411. roket();
412. petunjuk();
414. setcolor(10);
415. getch();
416. menu();
417. } **else** **if**(pilihan=="2"){
418. roket();
419. system("cls");
420. setcolor(8);
422. gotoxy (15,13); cout << "I want to play with ?";
423. gotoxy (15,15); cout << "1. Me  vs Komputer";
424. gotoxy (15,16); cout << "2. Me vs My Friend";
426. gotoxy (15,19); cout << " Pilihan Anda = ";
427. cin >> pemain;
429. **if** (pemain == "1"){
431. single();
432. }**else** **if** (pemain == "2"){
433. roket();
434. dobel();
435. }
436. **else** {
437. gotoxy(17,21); cout << "Sorry, this option doesn't exist";
438. setcolor (10);
439. menu();
440. }
441. setcolor (10);
442. menu();
443. }
444. **else** **if** (pilihan=="3"){
445. gotoxy(45,44); cout << "                                 ";
446. gotoxy(45,45); cout << "        \*\*FINISH, THANKYOU\*\*     ";
447. gotoxy(45,46); cout << "                                 ";
448. } **else** {
449. gotoxy (14,20); cout << "Sorry, this option doesn't exist ";
450. getch();
451. setcolor (10);
452. menu();
453. }
455. }
457. **void** loading (){
458. **char** a=177, b=219;
459. **for**(**int** i=0;i<=25;i++)
460. gotoxy(30,16);cout<<a;
462. Sleep(200);
463. cout<<"\r";
464. cout<<"\t\t\t\t\t ";
465. gotoxy (17,5);cout<<"      ÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»";cout<<endl;
466. gotoxy (17,6);cout<<"      ºÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛº";cout<<endl;
467. gotoxy (17,7);cout<<"  ÉÍÍÍ¼ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÈÍÍÍ»";cout<<endl;
468. gotoxy (17,8); cout<<" É¼ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÈ»";cout<<endl;
469. gotoxy (17,9);cout<<" ºÛÛÛÛÛÛÛÛÛÛÛÛÛÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»ÛÛÛÛÛÛÛÛÛÛÛÛÛº";cout<<endl;
470. gotoxy (17,10);cout<<" ºÛÛÛÛÛÛÛÛÛÛÛÛÛº                                                ºÛÛÛÛÛÛÛÛÛÛÛÛÛº";cout<<endl;
471. gotoxy (17,11);cout<<" ºÛÛÛÛÉÍÍÍ»ÛÛÛÛº                                                ºÛÛÛÛÉÍÍÍ»ÛÛÛÛº";cout<<endl;
472. gotoxy (17,12);cout<<" ºÛÛÛÛº   ºÛÛÛÛº                                                ºÛÛÛÛº   ºÛÛÛÛº";cout<<endl;
473. gotoxy (17,13);cout<<" ºÛÉÍÍ¼   ÈÍÍ»Ûº                                                ºÛÉÍÍ¼   ÈÍÍ»Ûº";cout<<endl;
474. gotoxy (17,14);cout<<" ºÛº         ºÛº   HANGMAN Loading....              ºÛº         ºÛº";cout<<endl;
475. gotoxy (17,15);cout<<" ºÛº         ºÛº                                                ºÛº         ºÛº";cout<<endl;
476. gotoxy (17,16);cout<<" ºÛÈÍÍ»   ÉÍÍ¼Ûº                                                ºÛÈÍÍ»   ÉÍÍ¼Ûº";cout<<endl;
477. gotoxy (17,17);cout<<" ºÛÛÛÛº   ºÛÛÛÛº                                                ºÛÛÛÛº   ºÛÛÛÛº";cout<<endl;
478. gotoxy (17,18);cout<<" ºÛÛÛÛÈÍÍÍ¼ÛÛÛÛº                                                ºÛÛÛÛÈÍÍÍ¼ÛÛÛÛº";cout<<endl;
479. gotoxy (17,19);cout<<" ºÛÛÛÛÛÛÛÛÛÛÛÛÛº                                                ºÛÛÛÛÛÛÛÛÛÛÛÛÛº";cout<<endl;
480. gotoxy (17,20);cout<<" ºÛÛÛÛÛÛÛÛÛÛÛÛÛº                                                ºÛÛÛÛÛÛÛÛÛÛÛÛÛº";cout<<endl;
481. gotoxy (17,21);cout<<" ºÛÛÛÛÛÛÛÛÛÛÛÛÛÈÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ¼ÛÛÛÛÛÛÛÛÛÛÛÛÛº";cout<<endl;
482. gotoxy (17,22);cout<<" È»ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÉ¼";cout<<endl;
483. gotoxy (17,23);cout<<"  ÈÍÍÍ»ÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÉÍÍÍ¼";cout<<endl;
484. gotoxy (17,24);cout<<"      ºÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛÛº";cout<<endl;
485. gotoxy (17,25);cout<<"      ÈÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ¼";cout<<endl;
486. gotoxy(45,16);
488. **for**(**int** i=0;i<=25;i++){
489. cout<<b;
490. Sleep(200);
492. }
493. }
495. **int** main (){
497. setcolor(2);

500. gotoxy (30,15); loading();
501. menu();
503. **return** 0;
505. }

# **BAB IV KESIMPULAN DAN SARAN**

## 4.1 Kesimpulan

Kesimpulan yang dapat kami ambil dari *game hangman* ini adalah *game* ini dibuat sesederhana mungkin agar memudahkan pemain dalam memainkan *game* ini. *Game* ini di lengkapi dengan petunjuk permainan, skor yang telah diperoleh pemain, dan dapat dimainkan dengan teman untuk mengisi waktu luang kita.

Dalam *game* ini juga dilengkapi info huruf yang pernah salah di tebak, tujuannya agar pemain tidak memasukan huruf yang sama dan bertujuan untuk memudahkan pemain untuk menebak huruf selanjutnya.

## 4.2 Saran

Untuk kenyamanan para pemain, sebaiknya program dilengkapi fitur untuk *hint* agar para pemain yang tidak bisa menebak dapat mendapat bantuan, selain itu kurang bervariasinya kosakata dalam kamus membuat kata yang ditebak hanya itu-itu saja.

# **DAFTAR PUSTAKA**

(-, technology of computing, 2018)

(-, C++ > Computer Graphics Code Examples, 2018)

(-, Simple program to create a moving car in graphics, 2018)

# **LAMPIRAN**