

This is actually Noble D. Bell's Text Adventure Engine, with slight modifications by me. I am posting it here for accessibility purposes.

```
*** * TEXT ADVENTURE ENGINE *
' * RELEASE 1.0.0 - FREEWARE *
' * BY NOBLE D. BELL ([[http://www.noblebell.com%29|http://www.noblebell.com]]) *
' *
' * NOTICE: *
' * You are free to use this text game engine for any use. The author
, Noble D. Bell, is not responsible in any way *
' * for the use or misuse of this engine. If you do make changes or m
odifications to the code I require that you email *
' * me the revised code. *
' **

' REVISIONS:
' * February 5, 2006 - OFFICIAL RELEASE 1.0.0

' TO DO:
' * Add: LIGHT / EXTINGUISH commands for using a light source
' * Program: LEARN / CAST for magic spells

'
=====

=====
=====
=====
' PROGRAM STARTS HERE
'

=====
=====
=====
=====
' Every LB'er should know what this is for.
NOMAINWIN

' Setup the Window that will be used to play the game.
GOSUB [Setup.GUI]

' Setup global variables for True/False
GLOBAL TRUE, FALSE
TRUE = 1
FALSE = 0
```

```
' Setup global variables that will be used throughout the game.
' score - the current score for the player, turns - the number of turns the player has taken
' rm - the current room the player is in, maxinventory - the total number of items the player can pack
' inventorycount - the total number of items the player is carrying, OK - used as a flag
' maxhits - the max number of damage a player can take, hits - the total points of damage the player has taken
' defense - the number needed to hit a player, offense - the amount of damage a player can inflict
' gold - the amount of money the player has, maxmp - the total amount of magic the player can wield
' mp - the total amount of magic the player has left, armor$ - the type of armor the player is wearing
' weapon$ - the type of weapon the player is wielding, encounter - flag indicating a current battle
' lantern - a flag indicating if the lantern is on or off, needlight - a flag indicating if an area is dark or not
' noun - the number of the selected noun, verb - the number of the selected verb
' gametitle$ - the title of your game, gamedescription - the description of your game, crlf$ - carriage return/linefeed
```

```
GLOBAL score,turns,maxscore,rm,maxinventory,inventorycount,OK
GLOBAL maxhits,hits,defense,offense,gold,maxmp,mp,armor$,weapon$
GLOBAL encounter,lantern,needlight,noun,verb
GLOBAL gametitle$,gamedescription$,crlf$
```

```
' Special game variables and arrays
' rooms$(100) - This contains the description for each room in the game upto 100 rooms.
' roomProperties(100, 2) - Determines if the room is available for teleporting.
' exits(100,15) - This contains the visible exits in each room
' help$(100) - This contains some help or a clue for a particular room in the game.
' objects$(100) - Contains a list of objects the player can interact with in the game upto 100.
' objectproperties(100,7) - Contains a list of different properties about an object in the game.
' commands$(24) - Contains a list of commands (verbs) that are used in the game.
' objectdescriptions$(100) - Contains a list of descriptions about each object in the game.
```

```
'  
' exits parameters:  
' 1 - North, 2 - East, 3 - South, 4 - West, 5 - Up, 6 - Down : Holds  
connecting room number. Has 0 if not an exit.  
' 7 - 12 are flags for each exit. (See example for use)  
' 13 - 15 are not used at present time  
'  
' object properties parameters:  
' 1 - the room location the object resides in.  
' 2 - tells if the object is takeable (TRUE,FALSE)  
' 3 - if the object is a monster tells how many hitpoints it has  
' 4 - if the object is a monster tells how hard it is to hit it (1-20  
)  
' 5 - if the object is a monster tells how much damage it can inflict  
(4,6,8,10,12,20)  
' 6 - special flag  
' 7 - Object is a source of Light(lantern, e.g.)  
'  
' room properties:  
' 1 - Room can be teleported to. Values: 0 - Not Visited 1 - Visited,  
able to TP 2 - Never TP  
' 2 - Room needs light  
  
DIM rooms$(100),exits(100,15),help$(100),commands$(27)  
DIM objects$(100),objectproperties(100,7),objectdescriptions$(100)  
DIM roomProperties(100, 2)  
  
' Assign the crlf$ with the ascii characters that represent a carriage  
e return and a linefeed.  
crlf$ = chr$(13)+chr$(10)  
  
' Initialize everything and begin  
GOSUB [Setup.General]  
GOSUB [Setup.Rooms]  
GOSUB [Setup.Objects]  
GOSUB [Begin]  
  
GOSUB [FleshRoom]  
GOTO [Display]  
  
'  
*** * WAIT FOR THE USER TO DO SOMETHING. *  
***
```

```
[EventLoop]
#m.txtCommand, ""
#m.txtCommand, "!setfocus"
Wait

'
**
' * THE PARSER - BREAKS DOWN THE COMMAND INTO A VERB + NOUN COMBINATIO
N. *
' ***
[Parser]
turns = turns + 1 ' add one to the turns variable.
#m.txtCommand, "!contents? command$";
' get what the user typed into the command text box.
command$ = upper$(command$)
' convert what the user typed to upper case text.
verb$ = "" ' clear the noun and verb strings to nil.
noun$ = ""
verb$ = WORD$(command$,1)
' using lb's word$ get the verb part of the command and put it in verb
$.
noun$ = WORD$(command$,2)
' using lb's word$ get the noun part of the command and put it in noun
$.
verb = 0 ' set the verb and noun id's to 0.
noun = 0

Gosub [Parser.Verb]
' check to see if the verb the user typed is one that we understand.
Return ' return to calling module.

[Parser.Verb]
' see if the verb the user typed is one that we understand and if it i
s
for x = 1 to 27
' then assign it it's action value (place in the array). if it is not
then
if verb$ = commands$(x) then
' set the verb id to 0 meaning not known.
verb=x
exit for
end if
next x

if verb = 0 then
' the verb is not known so tell the user to try again.
```

```
txt$ =
"I don't understand that command. Try saying it a different way please
."
Return
end if

select case verb
' the verb is known so let's act on it and see what we need to do with
it.
case 1 ' North
if exits(rm,1)<>0 then
GOSUB [Logic]
if OK=TRUE then
rm=exits(rm,1)
GOSUB [FleshRoom]
end if
else
txt$ = "You cannot go that direction."
end if

case 2 ' East
if exits(rm,2)<>0 then
GOSUB [Logic]
if OK=TRUE then
rm=exits(rm,2)
GOSUB [FleshRoom]
end if
else
txt$ = "You cannot go that direction."
end if

case 3 ' South
if exits(rm,3)<>0 then
GOSUB [Logic]
if OK=TRUE then
rm=exits(rm,3)
GOSUB [FleshRoom]
end if
else
txt$ = "You cannot go that direction."
end if

case 4 ' West
if exits(rm,4)<>0 then
GOSUB [Logic]
if OK=TRUE then
```

```
rm=exits(rm,4)
GOSUB [FleshRoom]
end if
else
txt$ = "You cannot go that direction."
end if

case 5 ' Up
if exits(rm,5)<>0 then
GOSUB [Logic]
if OK=TRUE then
rm=exits(rm,5)
GOSUB [FleshRoom]
end if
else
txt$ = "You cannot go that direction."
end if

case 6 ' Down
if exits(rm,6)<>0 then
GOSUB [Logic]
if OK=TRUE then
rm=exits(rm,6)
GOSUB [FleshRoom]
end if
else
txt$ = "You cannot go that direction."
end if

case 7 ' Save
if encounter = TRUE then
txt$ = "You cannot save a game during an encounter."
else
open "gamedata.dat" for output as #1
print #1,score
print #1,turns
print #1,rm
print #1,inventorycount
print #1,hits
print #1,defense
print #1,offense
print #1,gold
print #1,mp
print #1,armor$
print #1,weapon$
for x=1 to 100
```

```
for y=1 to 7
print #1,objectproperties(x,y)
next y
next x
for x=1 to 100
for y=1 to 15
print #1,exits(x,y)
next y
next x
for x=1 to 100
for y = 1 to 2
print #1, roomProperties(x, y)
next y
next x
print #1,lantern
close #1
txt$ = "Game saved."
end if

case 8 ' Load
if encounter = TRUE then
txt$ = "You cannot load a game during an encounter."
else
dim info$(10, 10)
if fileExists(DefaultDir$, "gamedata.dat") then
open "gamedata.dat" for input as #1
input #1,score
input #1,turns
input #1,rm
input #1,inventorycount
input #1,hits
input #1,defense
input #1,offense
input #1,gold
input #1,mp
input #1,armor$
input #1,weapon$
for x=1 to 100
for y=1 to 7
input #1,objectproperties(x,y)
next y
next x
for x=1 to 100
for y=1 to 15
input #1,exits(x,y)
next y
```

```
next x
for x=1 to 100
for y = 1 to 2
input #1, roomProperties(x, y)
next y
next x
input #1,lantern
close #1
txt$ = "Game loaded."
else
txt$ = "You don't have a game saved. Save one first."
end if
end if

case 9 ' Look
#m.txtView, ""
GOSUB [FleshRoom]

case 10 ' Inventory
txt$ = "You are carrying the following items:"+chr$(13)+chr$(10)
tmp = 0
for x = 1 to 100
if objectproperties(x,1)=0 then
if objects$(x)<>"" then
txt$ = txt$ + objects$(x)+chr$(13)+chr$(10)
tmp = 1
end if
end if
next x
if tmp
=0 then txt$=
txt$ + "Nothing.

case 11 ' Score
txt$ = "Your score is " + str$(score) + " out of "
+ str$(maxscore) + " in " + str$(turns) + " turns."+chr$(13)+chr$(10)
txt$ = txt$ + "You have " + str$(hits) + "/"
+ str$(maxhits) + " hit points." + chr$(13)+chr$(10)
txt$ = txt$ + "You have " + str$(mp) + "/"
+ str$(maxmp) + " magic points." + chr$(13)+chr$(10)
txt$ = txt$ + "You can inflict upto "
+ str$(offense) + " points of damage."+chr$(13)+chr$(10)
txt$ = txt$ + "Your defense is "
+ str$(defense) + ". "+chr$(13)+chr$(10)
txt$ = txt$ + "You have " + str$(gold) + " gold."+chr$(13)+chr$(10)
txt$ = txt$ + "You are wearing " + armor$ + " armor."
```

```
+chr$(13)+chr$(10)
txt$ = txt$ + "You are wielding " + weapon$ + "."

case 12 ' Inspect
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)=0 or objectproperties(noun,1)=rm then
txt$ = objectdescriptions$(noun)
GOSUB [Logic]
else
txt$ = "You don't see that here."
end if
end if

case 13 ' Use
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)<>0 then
txt$ = "You don't have that."
else
GOSUB [Logic]
end if
end if

case 14 ' Take
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)<>rm then
txt$ = "You don't see that here."
return
end if
if objectproperties(noun,1)=0 then
txt$ = "You already have that."
return
end if
if objectproperties(noun,2)=FALSE then
txt$ = "You cannot take that."
return
end if
if inventorycount+1 > maxinventory then
txt$ = "You cannot carry that, you are too heavy."
return
end if
objectproperties(noun,1)=0
txt$ = "Taken."
inventorycount = inventorycount + 1
```

```
end if

case 15 ' Drop
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)=0 then
objectproperties(noun,1)=rm
txt$ = "Dropped."
inventorycount = inventorycount - 1
else
txt$ = "You don't have that."
end if
end if

case 16 ' Fight
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)=rm then
if objectproperties(noun,3)=0 then
txt$ = "That is harmless. Why would you want to fight that?"
else
GOSUB [Battle]
end if
else
txt$ = "You don't see that here."
end if
end if

case 17 ' Quit
GOTO [Quit]

case 18 ' Help
txt$ = help$(rm)

case 19 ' Arm
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)<>0 then
txt$="You don't have that."
return
end if
if objects$(noun)=weapon$ then
txt$="You are already wielding that."
return
end if
weapon$ = objects$(noun)
```

```
offense = objectproperties(noun,5)
txt$ = "You are now wielding a " + weapon$ + "."
end if

case 20 ' UnArm
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)<>0 then
txt$="You don't have that."
return
end if
if objects$(noun)<>weapon$ then
txt$="You are not wielding that."
return
end if
weapon$ = "hands"
offense=4
txt$ = "You are now wielding " + weapon$ + "."
end if

case 21 ' Wear
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)<>0 then
txt$="You don't have that."
return
end if
if objects$(noun)=armor$ then
txt$="You are already wearing that."
return
end if
armor$ = objects$(noun)
defense = objectproperties(noun,4)
txt$ = "You are now wearing " + armor$ + " armor."
end if

case 22 ' Remove
GOSUB [Parser.Noun]
if noun <> 0 then
if objectproperties(noun,1)<>0 then
txt$="You don't have that."
return
end if
if objects$(noun)<>armor$ then
txt$="You are not wearing that."
return
end if
```

```
armor$ = "Cloth"
defense = 9
txt$ = "You are now wearing " + armor$ + " armor."
end if
case 23 ' Learn
txt$ = "Not used in this adventure."
case 24 ' Cast
txt$ = "Not used in this adventure."
case 25 ' Teleport
GOSUB [Logic]
If OK=TRUE and TOK = TRUE then
rm = val(noun$)
gosub [FleshRoom]
Else
txt$ = "Cannot TELEPORT there."
End If
case 26 ' Light
GOSUB [Parser.Noun]
If objectproperties(x, 7) = 1 then
If lantern then
txt$ = "You already have something lit!"
Else
lantern = 1
txt$ = "You have lit the ";objects$(x)
End If
Else
txt$ = "That cannot be lit."
End If
case 27 ' Extinguish
GOSUB [Parser.Noun]
If objectproperties(x, 7) = 1 then
If lantern then
lantern = 0
txt$ = "The ";objects$(x);" was put out."
Else
txt$ = "The ";objects$(x);" is not lit."
End If
Else
txt$ = "That cannot be extinguished."
End If

End Select

Return

[Parser.Noun]
```

```
' look to see if we understand what the object or noun is that the
for x = 1 to 18
' user entered. if we do understand it then assign it it's action
if noun$ = objects$(x) then
' id (place in the array) otherwise assign the noun id a 0 meaning
noun=x ' we did not know that object.
exit for
end if
next x

if noun = 0 then
txt$ = "I don't know what that is."
Return
end if

Return

'
*** * USER TAPS OK OR HITS ENTER, ACT ON THEIR COMMANDS. *
***

[Okay]
GOSUB [Parser]
GOTO [Display]

'

*** * DISPLAY ACTION *
***

[Display]
#m.txtView, "!contents? tmpTxt$";
#m.txtView, ""
'tmpTxt$=""
tmpTxt$ = tmpTxt$ + chr$(13) + chr$(10)
tmpTxt$ = tmpTxt$ + "> "
+ command$ + chr$(13) + chr$(10) + chr$(13) + chr$(10)
tmpTxt$ = tmpTxt$ + txt$ + chr$(13) + chr$(10)

if needlight
=TRUE and lantern=
FALSE then
tmpTxt$ =
"You cannot see anything. It is pitch black here."+chr$(13)+chr$(10)
end if

#m.txtView, "!setfocus"
#m.txtView, tmpTxt$
```

```
Call VerticalScroll hWnd(#m.txtView), _SB_BOTTOM ' Thanks Janet!!!
' Call VerticalScroll hWnd(#main.textbox), _SB_TOP
' Call VerticalScroll hWnd(#main.textbox), _SB_PAGEDOWN
' Call VerticalScroll hWnd(#main.textbox), _SB_PAGEUP

GOTO [EventLoop]

'

*** * DESCRIBE ROOM FOR DISPLAY ROUTINE *
***

[FleshRoom]
If roomProperties(rm, 1)
=0 then roomProperties(rm, 1)=
1 ' Allows TELEPORTING
If not(lantern) AND roomProperties(rm, 2) then
txt$ =
"The darkness is so blinding you can't see your hand in front of your
face."
txt$ = txt$ + chr$(13) + chr$(10)
) +
"If only you had a light source...
+ chr$(13)+chr$(10)+chr$(13)+chr$(10)
Else
txt$=rooms$(rm) +
chr$(13)+chr$(10)+chr$(13)+chr$(10)
) ' Let the program display the room's contents and it's
End If
txt$ = txt$ + "Exits: " ' visible exits along with the description.
tmp = 0
if exits(rm,1)<>0 then
txt$ = txt$ + "North,"
tmp=1
end if
if exits(rm,2)<>0 then
txt$ = txt$ + "East,"
tmp=1
end if
if exits(rm,3)<>0 then
txt$ = txt$ + "South,"
tmp=1
end if
if exits(rm,4)<>0 then
txt$ = txt$ + "West,"
tmp=1
```

```
end if
if exists(rm,5)<>0 then
txt$ = txt$ + "Up,"
tmp=1
end if
if exists(rm,6)<>0 then
txt$ = txt$ + "Down,"
tmp=1
end if

if right$(txt$,1)="" then
txt$ = left$(txt$,len(txt$)-1)
end if

if tmp
=0 then txt$=
txt$ + "None"

txt$ = txt$ + chr$(13)+chr$(10)+chr$(13)+chr$(10)
txt$ = txt$ + "You also see: " +chr$(13)+chr$(10)
tmp = 0
for x=1 to 100
if objectproperties(x,1)=rm then
txt$ = txt$ + objects$(x)+chr$(13)+chr$(10)
tmp=1
end if
next x
if tmp
=0 then txt$=
txt$+"Nothing"
txt$ = txt$+chr$(13)+chr$(10)
Return

'
*** * QUIT THE GAME *
 ***
[Quit]
CONFIRM gametitle$+chr$(13)+chr$(13)
"Are you sure you really want to quit?";answer$
if answer$="no" then GOTO [EventLoop]

txt$ =
"Thank you for playing. I really hope that you had a wonderful and ent
ertaining experience."+chr$(13)+chr$(10)
txt$ = txt$ +
```

```
"If you have any comments or questions then please visit my website at
the following"+chr$(13)+chr$(10)
txt$ = txt$ +
"location: http://www.noblebell.co
m"+chr$(13)+chr$(10)+chr$(13)+chr$(10)

NOTICE gametitle$+chr$(13)+txt$

Close #m
END

'

*** * FUNCTIONS *
***

' Specify the number of sides on the die and the number of times to r
oll and the result will be returned.
FUNCTION RollDice(sides,times)
tmp = 0
FOR x = 1 to times
tmp = tmp + int(rnd(1)*sides)+1
NEXT x
RollDice = tmp
END FUNCTION

' Checks to see if a file that is passed actually exists on the users
directory path specified.
FUNCTION fileExists(path$, filename$)
files path$, filename$, info$()
fileExists = val(info$(0, 0)) 'non zero is true
END FUNCTION

' Used to scroll to the bottom of the screen in the textbox.
SUB VerticalScroll handle, ScrollAmount
CallDll #user32, "SendMessageA", _
handle as Ulong, _
_WM_VSCROLL as Long, _
ScrollAmount as Long, _
0 as Long, _
result as Long
END SUB

'

*** * GUI SETUP *
***
```

```
[Setup.GUI]
ForegroundColor$ = "black"
BackgroundColor$ = "buttonface"
TextboxColor$ = "white"
TexteditorColor$ = "white"
ComboboxColor$ = "white"
ListboxColor$ = "white"

WindowWidth
=792 : WindowHeight=
595
UpperLeftX=int((DisplayWidth-WindowWidth)/2)
UpperLeftY=int((DisplayHeight-WindowHeight)/2)

textbox #m.txtView, 8, 10, 768, 515
textbox #m.txtCommand, 8, 530, 672, 25
stylebits #m.btnExit, _BS_MULTILINE, 0, 0, 0
button #m.default, "GO", [Okay], UL, 688, 530, 88, 25

stylebits #m.txtView, _ES_MULTIL
INE or _ES_READONLY or _WS_VSCROLL, _ES_AUTOHSCROLL , 0, 0

open "TEXT ADVENTURE" for dialog as #m

#m.txtView "!font arial 11"
#m.txtCommand "!font arial 11 bold"
#m.default "!font arial 11 bold"
#m "trapclose [Quit]"
#m.txtCommand, "!setfocus"

Return

'

*** * GENERAL *
***

[Setup.General]
' assign starting values to certain variables
score
=0 : maxscore=
500 ' current score is 0 and the maximim score can be only 500
turns
=0 : rm=
1 ' total turns taken is 0 and the starting room location is 1
maxinventory
```

```
=10 : inventorycount=
0
' total items that can be carried at once is 10 and they are carrying
0

maxhits = 25 ' the total number of hits you can take before you die.
hits = 25 ' you have 25 hits left before you die.
defense = 9
' (1-20) : 20-Easy to hit, 1-Near impossible to hit, 0-Can't hit!
offense = 4 ' the amount of damage that can be caused (4,8,10,12,20)
gold = 50 ' starting amount of money the player has
maxmp = 15 ' total magic points the player can spend
mp = 15 ' total magic points the player has left to spend

encounter = FALSE
' tells if the player is engaged in a battle (TRUE/FALSE)
lantern = FALSE
' tells if the player's lantern is lit or not (TRUE/FALSE)
needlight = FALSE
' tells if the room needs to have a light or not (TRUE/FALSE)
noun = 0 ' noun action id
verb = 0 ' verb action id
OK = FALSE ' flag (TRUE/FALSE)

' Using anything different from the items below they must be listed as a game object that the user can interact with.
' If the object is to be armor then you need to set the objectproperties defense parameter to the value you want.
' If the object is to be a weapon then you need to set the objectproperties damage parameter to the value you want.

armor$="CLOTH" ' Starting armor the player is wearing.
weapon$="HANDS" ' Starting weapon the player is using.

' Add the known commands to the commands$ array. Currently there are 24 known commands. The commands must be
' in UPPER case and only one word. You can add new commands by adding another element to the commands$ array
' with the command and then you will have to add the new element to the [Parser.Verb] module and also to the
' [Logic] module if needed.
commands$(1)="NORTH":commands$(2)="EAST":commands$(3)="SOUTH"
:commands$(4)="WEST":commands$(5)="UP":commands$(6)="DOWN"
commands$(7)="SAVE":commands$(8)="LOAD":commands$(9)="LOOK"
:commands$(10)="INVENTORY":commands$(11)="SCORE"
```

```
commands$(12) = "INSPECT" : commands$(13) = "USE"
: commands$(14) = "TAKE" : commands$(15) = "DROP" : commands$(16) = "FIGHT"
commands$(17) = "QUIT" : commands$(18) = "HELP"
: commands$(19) = "ARM" : commands$(20) = "UNARM"
: commands$(21) = "WEAR" : commands$(22) = "REMOVE"
commands$(23) = "LEARN" : commands$(24) = "CAST"
: commands$(25) = "TELEPORT" : commands$(26) = "LIGHT"
: commands$(27) = "EXTINGUISH"

Return

'
*** * ROOM DESCRIPTIONS, EXITS, ETC. *
***

[Setup.Rooms]
for x=1 to 100
help$(x) = "Try working it out yourself a little longer."
next x

for x=1 to 100
for y = 1 to 15
exits(x,y)=0
next y
next x

for x=1 to 100
rooms$(x) = ""
next x

rooms$(1) = "BOTTOM OF MOUNTAIN"+crlf$+crlf$
rooms$(1) = rooms$(1) +
"You are standing at the foot of what appears to be a really large mou
ntain."+crlf$
rooms$(1) = rooms$(1) +
"The rocks are too steep and slimy to climb. There is however a little
winding trail"+crlf$
rooms$(1) = rooms$(1) +
"that appears to be leading up into the mountain. There is a heavy mist
in the air."

rooms$(2) = "GAP IN THE PATH"+crlf$+crlf$
rooms$(2) = rooms$(2) +
"There is a small gap in the pathway here. It appears to be man-
made. The air is starting"+crlf$
rooms$(2) = rooms$(2) +
"to become more chilled and thin. The mist from earlier has all but di
```

sapated."

```
rooms$(3) = "MOUNTAIN PASS"+crlf$+crlf$  
rooms$(3) = rooms$(3) +  
"This is a leveled out area that looks to be a simple camp site. There  
is nothing useful here though."  
  
rooms$(4) = "HOLE IN MOUNTAIN SIDE"+crlf$+crlf$  
rooms$(4) = rooms$(4) +  
"There is an opening inside one of the large rocks here. It appears to  
be large enough"+crlf$  
rooms$(4) = rooms$(4) +  
"for a person to get inside. It is aweful dark looking in there and yo  
u can hear faint"+crlf$  
rooms$(4) = rooms$(4) + "sounds of dripping water in the distance."  
  
rooms$(5) = "SPLIT IN THE PATH"+crlf$+crlf$  
rooms$(5) = rooms$(5) +  
"The cavern that you are in seems to split into two different directio  
ns here."+crlf$  
rooms$(5) = rooms$(5) +  
"There is a strange wooden door on the north side of this passage and  
an opening"+crlf$  
rooms$(5) = rooms$(5) + "leading off into darkness toward the south."  
  
rooms$(6) = "LOST TOMB"+crlf$+crlf$  
rooms$(6) = rooms$(6) +  
"This is an oblong chamber where someone or something burries others.  
Everything is a mess"+crlf$  
rooms$(6) = rooms$(6) +  
"and all the tombs are open with nothing inside. Almost as if the cont  
ents were stolen."  
  
rooms$(7) = "WEST TOMB CHAMBER"+crlf$+crlf$  
rooms$(7) = rooms$(7) +  
"This area of the tomb is scattered with shrouds and busted open tombs  
. "  
  
rooms$(8) = "EAST TOMB CHAMBER"+crlf$+crlf$  
rooms$(8) = rooms$(8) +  
"This area of the elongated tomb is covered with busted open tombs."  
  
rooms$(9) = "WEST OF RAVINE"+crlf$+crlf$  
rooms$(9) = rooms$(9) +  
"You are standing just to the west of a very large and deep ravine car
```

```
ved into the mountain"+crlf$  
rooms$(9) = rooms$(9) +  
"surface. It kind of looks like this is where bad things have taken place in the past."
```

```
rooms$(10) = "EAST OF RAVINE"+crlf$+crlf$  
rooms$(10) = rooms$(10) +  
"The ravine is to the west of you. Things seem oddly different here. More so than normal."
```

```
rooms$(11) = "BONE CHAMBER"+crlf$+crlf$  
rooms$(11) = rooms$(11) +  
"The floor in this area is completely covered in unidentifiable bones of all shapes and sizes."
```

```
rooms$(12) = "SWIRLING POOL"+crlf$+crlf$  
rooms$(12) = rooms$(12) +  
"Standing in the center of this odd and foul-smelling pool of liquid stands a very unique"+crlf$  
rooms$(12) = rooms$(12) +  
"statue. There is a magical disturbance in the air. Some spells may be affected."
```

```
exits(1,1) = 2  
exits(2,1) = 3  
exits(2,3) = 1  
exits(2,2) = 4  
exits(3,3) = 2  
exits(4,4) = 2  
exits(4,2) = 5  
exits(5,4) = 4  
exits(5,3) = 6  
exits(5,1) = 9  
exits(5,7) = 1 'locked  
exits(6,1) = 5  
exits(6,2) = 8  
exits(6,4) = 7  
exits(7,2) = 6  
exits(8,4) = 6  
exits(9,2) = 10  
exits(9,3) = 5  
exits(9,8) = 1 'ravine  
exits(10,1) = 12  
exits(10,3) = 11
```

```
exits(10,4) = 9
exits(11,1) = 10
exits(12,3) = 10
roomProperties(12, 1) = 2 'Can't TP to or from room 12
roomProperties(12, 2) = 1 'Need light

Return

'
*** * OBJECTS, OBJECT PROPERTIES, OBJECT DESCRIPTIONS *
***

[Setup.Objects]
for x = 1 to 100
objects$(x)=""
next x

objects$(1) = "ORC"
objectproperties(1,1)=3
objectproperties(1,2)=FALSE
objectproperties(1,3)=6
objectproperties(1,4)=8
objectproperties(1,5)=4
objectdescriptions$(1) = "A massive frame, pig-
like head, pale green. Very unfriendly looking."

objects$(2) = "SKELETON"
objectproperties(2,1)=6
objectproperties(2,2)=FALSE
objectproperties(2,3)=4
objectproperties(2,4)=9
objectproperties(2,5)=4
objectdescriptions$(2) =
"An animated corpse of bones wielding a dagger."

objects$(3) = "SWORD"
objectproperties(3,1)=7
objectproperties(3,2)=TRUE
objectproperties(3,5)= 8
objectdescriptions$(3) =
"A short sword made of finely crafted steel with a jewel-enlaid hilt."

objects$(4) = "ROPE"
objectproperties(4,1)=7
objectproperties(4,2)=TRUE
objectdescriptions$(4) = "50' of coiled rope"
```

```
objects$(5) = "KEY"
objectproperties(5,1)=8
objectproperties(5,2)=TRUE
objectdescriptions$(5) = "A rusted-
metal skeleton key. It might be used to unlock more than one door."

objects$(6) = "STATUE"
objectproperties(6,1)=12
objectproperties(6,2)=FALSE
objectdescriptions$(6) =
"This is just a simple stone statue .. Wait! You feel funny and appear
to be feeling better."

objects$(7) = "LANTERN"
objectproperties(7, 1) = 1
objectproperties(7, 2) = TRUE
objectproperties(7, 7) = TRUE
objectdescriptions$(7) = "A rather old lantern, but appears to work."

'
*** * GAME LOGIC BELOW *
***

[Logic]
OK = TRUE
TOK = TRUE

' Once a fight starts you cannot run from it. You must finish it.
if encounter = TRUE then
if verb=1 or verb=2 or verb=3 or verb=4 or verb=5 or verb=6 then
txt$ = "You cannot run!"
OK = FALSE
return
end if
end if

'Checks if you can TELEPORT to a room
roomnum = val(noun$)
If roomnum > 12 or roomnum < 1 then TOK = FALSE
If roomProperties(roomnum, 1)
=0 or roomProperties(roomnum, 1)=
2 then TOK = FALSE
```

```
If roomProperties(rm, 1)
=2 then TOK=
  FALSE

  ' Special things that happen in selected rooms.
  select case rm
  case 12
    if verb
    =12 and noun=
    6 then
      txt$ = objectdescriptions$(6)+crlf$
      txt$ = txt$ +
      "Poof! All your damage has been healed, and your magic restored."
      hits = maxhits
      mp = maxmp
    end if
    case 5
      if verb
      =1 and exits(5,7)=
      1 then
        txt$ = "That doorway appears to be locked or something."
        OK=FALSE
      end if
      if verb
      =1 and exits(5,7)=
      0 then
        OK=TRUE
      end if
      if verb
      =13 and noun=
      5 then
        exits(5,7)=0
        txt$ = "The door has been unlocked."
      else
        txt$ = "The door is locked."
      end if
      case 9
        if verb
        =2 and exits(9,8)=
        1 then
          txt$ = "You can't jump that ravine. It is too wide."
          OK=FALSE
        end if
        if verb
        =2 and exits(9,8)=
        0 then
```

```
OK=TRUE
end if
if verb=13 and noun=4 then
txt$ =
"You string the rope through a hole in the ceiling so that you can swi
ng across."
exits(9,8)=0
else
txt$ = "You can't get across that way."
end if
end select

Return

'

*** * THE BATTLE ENGINE *
***

[Battle]
' We are in a battle so we cannot load or save a game.
encounter = TRUE

txt$ = ""
txt$ = txt$ + "You attack the "
+ objects$(noun) + " using your " + weapon$ + "..." +crlf$

' Player attacks the monster
toHit = RollDice(20,1)
if toHit <= objectproperties(noun,4) then
toDamage = RollDice(offense,1)
txt$ = txt$ + "You inflict " + str$(toDamage) +
" points of damage on the " + objects$(noun)+". "+crlf$
objectproperties(noun,3)=objectproperties(noun,3)-toDamage
if objectproperties(noun,3) <= 0 then
txt$ = txt$ + "You have slain the " + objects$(noun)+crlf$
objectproperties(noun,1) = -1
gp = RollDice(6,1) * 10
txt$ = txt$ + "You have found " + str$(gp) +
" gold coins." +crlf$+crlf$
gold=gold+gp
score = score + 5
encounter = FALSE
Return
end if
else
txt$ = txt$ + "You miss the " + objects$(noun)+" !!" +crlf$
```

```
end if
```

```
' Monster attacks the player
txt$ = txt$ + "The " + objects$(noun) + " attacks you..." +crlf$
toHit = RollDice(20,1)
if toHit <= defense then
  toDamage = RollDice(objectproperties(noun),5),1)
  txt$=txt$+"The " + objects$(noun)+" inflicts "
  + str$(toDamage) + " points of damage on you." +crlf$
  hits=hits-toDamage
  if hits <=0 then
    txt$ = txt$ + "You have been slain!" +crlf$
    close #m
  end
end if
else
  txt$ = txt$ + "The " + objects$(noun) + " , misses you." +crlf$
end if
```

```
Return
```

```
'  
*** * DISPLAY TITLE, INTRODUCTION, INSTRUCTIONS *  
***
```

```
[Begin]  
gametitle$ = "Mysterious Caverns 1.00"  
gamedescription$ = ""
```

```
txt$=""  
txt$ = txt$ + gametitle$ +crlf$  
txt$ = txt$ + "By: Noble D. Bell" +crlf$ +crlf$  
txt$ = txt$ + gamedescription$ +crlf$ +crlf$  
txt$ = txt$ +  
"To play this game you use text commands. The commands must be in the  
form of [verb][noun]. The" +crlf$  
txt$ = txt$ +  
"game understands 24 different commands and quite possibly several dif  
ferent nouns." +crlf$ +crlf$  
txt$ = txt$ +  
"Here are the commands that can be used in this game:" +crlf$ +crlf$
```

```
for x=1 to 24
txt$=txt$+commands$(x)+" , "
next x
txt$=left$(txt$,len(txt$)-1)
txt$ = txt$+crlf$+crlf$
txt$ = txt$ + "Just press [ENTER] to begin.

'notice gametitle$+crlf$+txt$
#m.txtView txt$
#m.txtCommand "LOOK"
wait
```

Return

```
'=====
=====
=====
' PROGRAM ENDS HERE
'
=====
=====
=====
```

[thedarkfreak](#) Jan 27, 2008(Chris Iverson)