

Więzienie

Gra przygodowa



Wstęp

„Zostałeś uwięziony, twoim celem będzie opuszczenie lochu pełnego zagrożeń i niebezpieczeństw. Czy wyjdiesz cało? Czy wyniesiesz ze sobą łupy? A może zginiesz od przypadkowego ciosu? Chcesz się dowiedzieć? Zagraj!”

Więzienie to gra przygodowa, w której zdobywasz przedmioty, pokonujesz przeszkody i starasz się wyjść z bagna do którego wpadłeś, nie wiedząc nawet dlaczego. Mało prawdopodobne abyś za pierwszym razem uciekł, ale nie poddawaj się, a w końcu ci się uda.

Instrukcja zawiera nie tylko podstawowe informacje o sterowaniu, ale też podstawy rozgrywki, statystyki przedmiotów, porady innych graczy oraz dodatkowe informacje.



Sterowanie

Co chwila gra pyta się co zamierzasz zrobić/ gdzie idziesz, wtedy należy wpisać cyfrę według poniższej instrukcji i nacisnąć enter.

Chodzenie

Komendy kierunkowe:

- „2” - dół
- „4” - lewo
- „6” - prawo
- „8” - góra
- „5” - otworenie ekwipunku/podniesienie przedmiotu

Ekwipunek

Pierw wpisujesz liczbę odpowiadającą za pozycje przedmiotu, a następnie podejmujesz akcję:

- „1” - wyrzucenie przedmiotu (nieodwracalne)
- „2” - czytanie opisu
- „3” - za ekwipowanie (o ile się da)
- „4” - Użycie przedmiotu(tylko mikstura lecznicza)

*Wpisanie liczby większej niż maksymalna pozycja kończy się zamknięciem plecaka

Walka

- „1” - Skupienie się na wyprowadzeniu ataku
- „2” - Skupienie się na uniknięciu wrogiego ciosu
- „3” - Wyprowadzenie silnego ciosu(zajmuję 2 tury)
- „4” - Wypicie mikstury leczniczej (o ile jeszcze jakąś posiadasz)

Rozgrywka

Z grubsza, gra dzieli się na dwa tryby, eksplorację i walkę. Umiejętne korzystanie z obu trybów skończy się uzyskaniem najlepszego wyniku.

Eksploracja

Więzienie nie jest puste, znajduje się tu wielu wrogów jak i pomocnych przedmiotów, warto je odróżniać:

- A,B,C... - Litery zawsze symbolizują wrogie postacie, kontakt z nimi rozpoczyna walkę.
- !,#,+,= - Reprezentują przedmioty, które po podniesieniu trafiają do plecaka.
- @ - Ikona symbolizująca położenie gracza, wraz z jej przemieszczeniem zmienia się widok okolicy.
- ^ - Taki znak zaznacza ważne miejsca, zwykle prowadzące do jakiegoś zakończenia.

Walka

Jeśli zostaniesz zaatakowany nie możesz uciec. W każdej swojej turze będziesz atakował, ale możesz dodatkowo skupić się na czymś jeszcze:

- Atak – Zamiast robić coś innego skupiasz się na wyprowadzeniu dobrego ciosu.
- Unik – Nie starasz się za bardzo trafić, zachowujesz odległość i przygotowujesz się do uniku.
- Silny cios – Wykonujesz potężny zamach i jeśli trafisz zadasz większe obrażenia, ale zajmują to dwie tury.
- Wypicie mikstury – Szybko sięgasz po medykament i go aplikujesz. Jeśli nie masz takowego w ekwipunku, to niestety marnujesz turę na szukaniu.

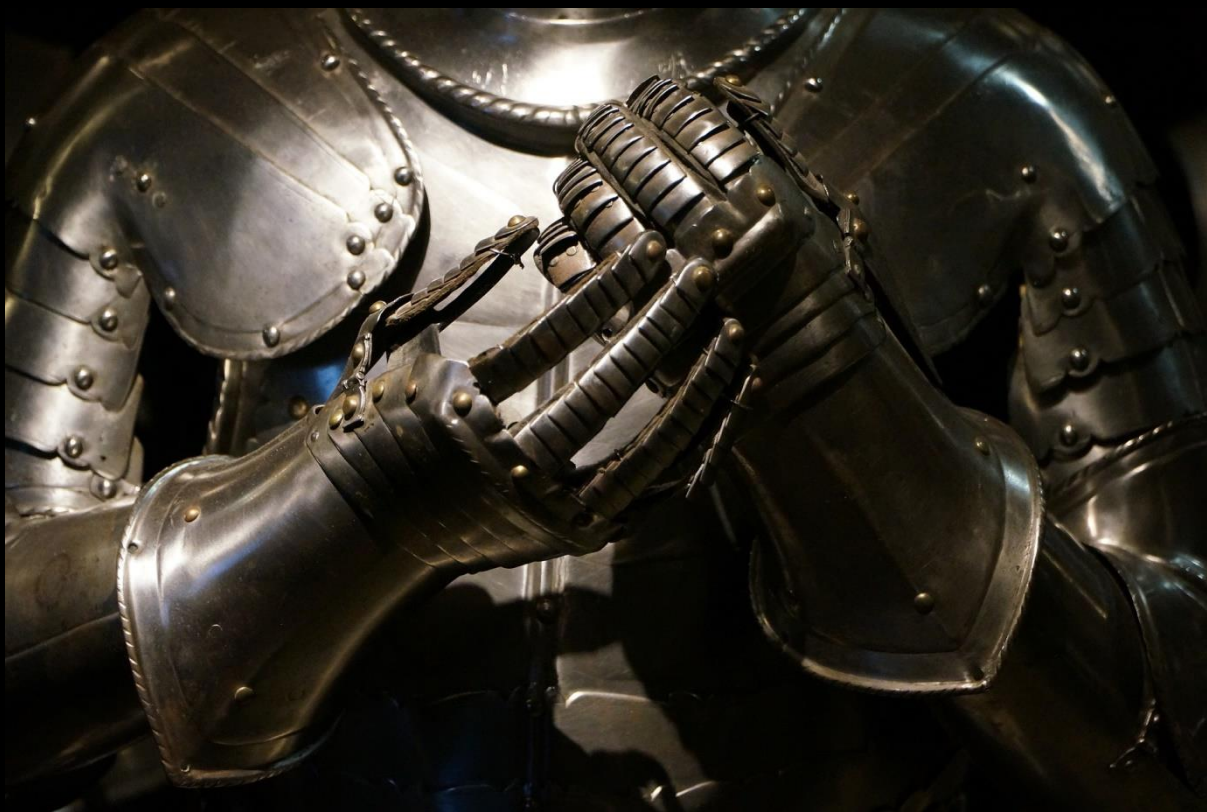
Przedmioty

W czasie gry możesz mieć za ekwipowane tylko dwa przedmioty, więc warto zerknąć poniżej na ich statystyki, aby łatwiej było wybrać swój ekwipunek.

| Nazwa | Atak | Unik | Obrażenia |
|-----------------|------|------|-----------|
| Sztylet | +5% | +0% | +0 |
| Miecz | +20% | +10% | +0 |
| Bicz | +0% | +0% | +0 |
| Sejmitar | +10% | +0% | +1 |
| Pancerz Płytowy | -5% | +30% | +0 |
| Skórznia | +0% | +15% | +0 |

Dodatkowe przedmioty:

- Mikstura lecznicza – Pozwala zregenerować życie bohatera do wartości 10.
- Notatki – Nie mają bezpośredniego wpływu na grę, natomiast można je czytać.



Porady/opinię testerów

- „Zdobyłeś coś nowego? Eksperymentuj! Nawet przy niewielu opcjach możesz natrafić na ciekawą kombinację ekwipunku.” Anonim
- „Naparzanie daje punkty” Anonim
- „Wyjdź poza opis. Być może delikatne zespolenie z postacią pozwoli odkryć coś jeszcze?” Szymon Teodorczyk
- „Nie używaj za wcześnie mikstur! Lepiej zaczekaj na najbardziej opłacalny moment.” Anonim
- „Nie klikać liter !!!!!” Tomasz Sroka
- „Walka nie zawsze jest najlepszym rozwiązaniem. Jeżeli możesz, nie ryzykuj utraty zdrowia walcząc z każdym napotkanym strażnikiem.” Anonim
- „W walce staraj się korzystać z magii matematyki, a okaże się, że nie jest aż tak niebezpieczna. Walka, nie matematyka...” Szymon Teodorczyk
- „Jeżeli masz szansę zdobyć przedmiot bez walki to ją wykorzystaj!” Anonim



Przy projekcie pracowali:

- Szymon Rogoziński – Projekt gry, mechanika i zasady, główny koder.
- Tomasz Sroka – Główna pomoc merytoryczna związana z programowaniem, tester.
- Szymon Teodorczyk – Naczelny redaktor tekstu, tester.
- Maciek Stępniaak – Tester.

Źródła

- Wiersz zawarty w grze nosi tytuł „Gniew”, został opublikowany pod tym linkiem (<http://wiersze.kobieta.pl/wiersze/gniew-457281>) przez autora o Nicku „3140”
- Zdjęcia w instrukcji (<https://pixabay.com>)

Kod źródłowy

```
# -*- coding: utf-8 -*-
import random, copy
# Klasy
class character:
    #celność/unik/obrażenia
    Bonus=[0,0,0]
    def __init__(self,name,acc,dodge,hp,dm,pause,notep,notek,movement,wort,icon):
        self.name=name
        self.acc=acc
        self.dodge=dodge
        self.hp=hp
        self.dm=dm
        self.pause=pause
        self.notep=notep
        self.notek=notek
        self.movement = movement
        self.wort=wort
        self.icon=icon

class item:
    def __init__(self,name,stat,notep,notek,slot,wort,icon):
        self.name = name
        self.stat=stat
        self.notep=notep
        self.notek=notek
        self.slot=slot
        self.wort = wort
        self.icon=icon

#Funkcje
def score():
    for i in range(len(Plecak)):
        gracz.wort+=Plecak[i].wort
    return "W trakcie gry zdobyłeś",gracz.wort,"Punktów"

def czyt(start,end):
    f=open("Text.txt","r+",encoding="utf-8")
    i=1
    for line in f.readlines():
        if i>=start and i<=end:
            print(line.rstrip())
        i+=1

def putin(pytanie):
    zm=""
    while zm=="":
        zm=input(pytanie)
    return int(zm)

def Torba(Stat,bag,eq):
    p=0
    k=0
    while p<len(bag):
        for i in range(0,len(bag)):
            if eq[bag[i].slot]!=0:
```



```

        if eq[bag[i].slot].name==bag[i].name:
            print(i,"*",bag[i].name)
        else:
            print(i, bag[i].name)
    else:
        print(i,bag[i].name)
p=int(input("który przedmiot wybierasz?"))
if p<=len(bag):
    k=int(input("Co zamierzasz zrobić?"))
    #1=wyrzucić,2=czytanie opisu przedmiotu,3=ekwipowanie,4=użycie
mikstury leczniczej
    if k==1:
        if bag[p] in eq:
            Stat.acc -= bag[p].stat[0]
            Stat.dodge -= bag[p].stat[1]
            Stat.dm -= bag[p].stat[2]
            eq[eq.index[bag[p]]] = 0
        del bag[p]
    elif k==2:
        czyt(bag[p].notep,bag[p].notek)
    elif k==3 and bag[p].slot!=0:
        if eq[bag[p].slot]!=0:
            Stat.acc+=bag[p].stat[0]- eq[bag[p].slot].stat[0]
            Stat.dodge+= bag[p].stat[1] - eq[bag[p].slot].stat[1]
            Stat.dm+= bag[p].stat[2] - eq[bag[p].slot].stat[2]
        else:
            Stat.acc+= bag[p].stat[0]
            Stat.dodge+= bag[p].stat[1]
            Stat.dm+= bag[p].stat[2]
            eq[bag[p].slot]=bag[p]
    elif k==4 and bag[p]==m1:
        del bag[p]
        Stat.hp=10
return Stat,bag,eq

def akcja(cl,a):
    cl.Bonus = [0, 0, 0]
    Zm=[4,6,1]
    Text=["Zamierza atakować","Zamierza unikać","Zamierza wykonać silny cios",
"Wypija miksturę uzdrawiającą", "Marnuję turę"]
    if a == 1 or a==2 or a==3:
        cl.Bonus[a-1] = Zm[a-1]
    if a==3:
        cl.pause=1
    if a==4:
        if m1 in Plecak:
            del Plecak[Plecak.index(m1)]
            cl.hp=10
        else:
            print("Nie masz mikstur")
            a=5
    if a<5:
        print(cl.name,Text[a-1])
    return cl

def test(poziom,b):
    wynik=random.randint(1,20)+b
    if wynik >=poziom:
        return True

```

```

    return False

def atak(atak,obrona):
    if atak.pause<1:
        c=test(obrona.dodge+obrona.Bonus[1],atak.Bonus[0])
        if c==True:
            obrona.hp-=atak.dm+atak.Bonus[2]
            print(obrona.name,"trafienie")
        else:
            print(obrona.name,"Atak uniknięty")
    return(atak,obrona)

def walka(atakujący,obrońca,p):
    print("Walka!")
    resthp=obrońca.hp+1
    czyt(obrońca.notep,obrońca.notek)
    while atakujący.hp>0 and obrońca.hp>0:
        print("Twoje życie",atakujący.hp)
        print("Życie wroga",obrońca.hp)
        if atakujący.pause==0:
            atakujący = akcja(atakujący,putin("Co robisz?"))
        else:
            atakujący.pause=0
        if obrońca.pause == 0:
            obrońca=akcja(obrońca,random.randint(1,3))
        else:
            obrońca.pause=0
        atakujący,obrońca=atak(atakujący,obrońca)
        if obrońca.hp>0:
            obrońca,atakujący=atak(obrońca,atakujący)
    if atakujący.hp>0:
        print("Wygrałeś")
        obrońca.hp=resthp-1
        if p>-1:
            if OBM[p]==wsp:
                OBM[p]=nw
            else:
                del OBM[p], OBMX[p]
        atakujący.wort+=obrońca.wort
        return True
    else:
        print("Przegrałeś")
        endings(1)
        return False

def move(Location,map):
    P=[0,1,0,0,-1,0,1,-1,0]
    a=0
    while a==0:
        T =Location.copy()
        a=putin("Gdzie idziesz?")
        if a==2 or a==4 or a==6 or a==8:
            T[0],T[1]=T[0]+P[a-1],T[1]+P[a]
            if map[T[0]][T[1]]=="|" or map[T[0]][T[1]]=="_":
                print("Ściana")
                a = 0
            elif T in OBMX :
                gracz.wort -= 1
            return T,2

```

```

        else:
            return T,0
    elif a == 5:
        return T,1
    else:
        a=0

def botmove(Location,map):
    P=[0,1,0,0,-1,0,1,-1,0]
    a=0
    while a==0:
        T =Location.copy()
        a=random.randint(1,4)*2
        T[0],T[1]=T[0]+P[a-1],T[1]+P[a]
        if map[T[0]][T[1]]=="_" or map[T[0]][T[1]]=="|":
            a=0
        else:
            return T

def warfog(położenie,mp,Mm):
    for i in range(4):
        if abs(Mm[i]-położenie[i//2])<4 or abs(Mm[i]-położenie[i//2])>11:
            if abs(Mm[i]-położenie[i//2]+1)==4 or abs(Mm[i]-położenie[i//2]+1)==11
:
                Mm[i]+=1
            else:
                Mm[i]-=1
    for j in range(len(mp)-1,-1,-1):
        if j>Mm[0] and j<Mm[1]:
            del mp[j][Mm[3]:len(mp[j])]
            if Mm[2]>=0:
                del mp[j][0:Mm[2]]
        else:
            del mp[j]
    return mp

def zagadka():
    if w1 in Plecak and w2 in Plecak and w3 in Plecak:
        print("Wydaje mi się, że trzeba ułożyć te notatki w całość.")
        print("Wybierz, który fragment, ma być, na którym miejscu")
        Zag=[]
        Odp=[w1,w2,w3]
        for j in range(len(Plecak)):
            if Plecak[j] in Odp:
                Zag.append(Plecak[j])
        while True:
            for j in range(3):
                print(j)
                czyt(Zag[j].notep,Zag[j].notek)
            for j in range(3):
                print("Pozycja",j)
                z=int(input("Wybierz odpowiedni fragment"))
                Zag[j],Zag[z]=Zag[z],Zag[j]
            if Zag[0]==Odp[0] and Zag[1]==Odp[1] and Zag[2]==Odp[2]:
                return 2
            else:
                print("Chyba coś mi się pomyliło.Spróbuję jeszcze raz")
    else:
        print("Chyba czegoś jeszcze nie posiadam")

```

```

        return 0
def endings(end):
    if end==1:
        czyt(99,103)
    elif end==2:
        czyt(95,97)
        if walka(gracz,ld,-1):
            czyt(105,112)
        else:
            endings(1)
    elif end==3:
        czyt(91, 93)
        if walka(gracz, de,-1):
            czyt(114, 117)
        else:
            endings(1)
    gracz.hp=0

#Początkowe zmienne
#obiekty
gracz=character(str(input("podaj swe imię")),0,4,6,1,0,0,0,0,0,'@')
wsp=character("Współwięzień",-1,2,3,1,0,30,32,3,30,'W')
st=character("Strażnik z kamienia",0,4,6,1,0,34,37,5,60,'S')
ld=character("Łowca demonów",3,8,6,1,0,39,43,0,250,'ł')
de=character("Demon z wnętrza",1,4,10,2,0,45,48,0,350,'D')
ml=item("Mikstura lecznicza",10,2,4,0,15,'+')
sz=item("Sztylet",[1,0,0],6,8,1,10,'!')
m=item("Miecz",[4,2,0],10,12,1,30,'!')
sj=item("Sejmitar",[2,0,1],14,16,1,70,'!')
b=item("Bicz",[1,4,0],18,20,1,70,'!')
pp=item("Pancerz płytowy],[-1,6,0],22,24,2,100,'=')
sk=item("Skórznia",[0,3,0],26,28,2,40,'=')
nw=item("Notatka więźnia",0,50,52,0,0,"#")
dn=item("Dziwna notatka",0,54,55,0,0,"#")
np=item("Notatka w pokoju",0,57,57,0,0,"#")
w1=item("Fragment wiersza",0,59,62,0,0,"#")
w2=item("Fragment wiersza",0,64,67,0,0,"#")
w3=item("Fragment wiersza",0,69,72,0,0,"#")
#Zmienne i tablice
Plecak=[sz]
Ekwipunek=[0,0,0]
YX=[23,14]
OBMYX=[[23,17],[15,13],[6,17],[19,30],[9,37],[10,55],[18,52],[32,48],[33,30],[38,2
3],[13,1],[17,1],[7,40],[13,60],[13,76],[21,46],[21,24],[37,16],[11,40],[25,39],[1
6,4],[7,60],[35,34],[39,16],[21,39],[35,53],[3,13],[7,51]]
OBM=[wsp,st,st,st,st,st,st,st,st,st,ml,ml,ml,ml,ml,ml,ml,ml,m,sj,b,pp,sk,dn,np,w1,
w2,w3]
ends=[[23,43],[0,71],[0,72]]
#0=minY,1=maxY,2=minX,3=maxX
Minmax=[YX[0]-6,YX[0]+6,YX[1]-6,YX[1]+6]
ReMap=[]
f=open("WięzienieMap.txt","r+",encoding="utf-8")
for line in f.readlines():
    line=list(line.rstrip())
    ReMap.append(line)
Map=copy.deepcopy(ReMap)
Map[YX[0]][YX[1]]=gracz.icon
Map=warfog(YX,Map,Minmax)
czyt(75,83)

```



```

cont=str(input(":"))
czyt(85,89)
cont=str(input("Wpisz cokolwiek aby zacząć"))
for i in range(len(Map)):
    print(*Map[i])
#Pętla rozrywki
while 1!=2:
    Map=copy.deepcopy(ReMap)
    YX,it = move(YX, Map)
    if YX in OBMX:
        point = OBMX.index(YX)
    if YX in ends:
        if YX[0] > 3:
            if zagadka()==2:
                endings(3)
            else:
                endings(2)
    if it==1:
        if YX in OBMX:
            Plecak.append(OBM[OBMX.index(YX)])
            print("Podnosisz",OBM[point].name)
            del OBM[point],OBMX[point]
        else:
            gracz, Plecak, Ekwipunek = Torba(gracz,Plecak,Ekwipunek)
    elif it==2 and type(OBM[point])==character:
        wynik=walka(gracz, OBM[point],point)
    Map[YX[0]][YX[1]] = gracz.icon
    for i in range(len(OBM)):
        MVMT=random.randint(0,10)
        if type(OBM[i])==character and MVMT<OBM[i].movement:
            OBMX[i]=botmove(OBMX[i],Map)
        Map[OBMX[i][0]][OBMX[i][1]] = OBM[i].icon
    if YX in OBMX:
        point = OBMX.index(YX)
        if type(OBM[point])==character:
            wynik=walka(gracz, OBM[point],point)
    if gracz.hp<=0:
        print(*score())
        break
    Map=warfog(YX,Map,Minmax)
    for i in range(len(Map)):
        print(*Map[i])

```