



AUTOMATIZAREA JOCULUI CANDY CRUSH IN PYTHON

Ciocan Gabriel



OBIECTIVE

- Crearea unui joc Candy Crush simplificat in Python.
- Automatizarea mutarilor pentru atingerea unui scor tinta.
- Inregistrarea rezultatelor si analiza statistica a jocurilor.

SETARI INITIALE

- **DIMENSIUNEA TABLEI:** 11X11
- **CULORI DISPONIBILE:** ROSU (R), GALBEN (Y), VERDE (G), ALBASTRU (B)
- **SCOR TINTA:** 10,000 PUNCTE
- **SIMBOLIZARE CULORI:** {0: ' ', 1: 'R', 2: 'Y', 3: 'G', 4: 'B'}

```
#setari pentru joc
```

```
RANDURI = 11
```

```
COLOANE = 11
```

```
CULORI = [1, 2, 3, 4] # 1=Rosu, 2=Galben, 3=Verde, 4=Albastru
```

```
SCOR_TINTA = 10000
```

```
SIMBOL_CULORI = {0: ' ', 1: 'R', 2: 'Y', 3: 'G', 4: 'B'}
```

INITIALIZAREA TABLEI

- TABLA ESTE GENERATA ALEATOR LA INCEPUT.
- SE ELIMINA AUTOMAT COMBINATIILE EXISTENTE (CASCADA INITIALA).
- FUNCTIE CHEIE: REZOLVA_TABLA_INITIALA()

Joc 0 initial:

```
R G R B Y G B G R B Y
B R Y Y R G B G G B Y
Y B Y B G B Y B R G B
Y Y G G R Y R Y R B G
R Y Y R R G R R G R B
Y G R B B G G R B Y B
Y B Y G Y R R Y G B Y
G Y G R R B Y B Y Y B
Y B G R G B Y R B R G
B B Y G G R R Y Y G B
B R Y G Y G Y R B B R
```

Swap: (4,0) <-> (4,1) | Scor: 50 | Swap-uri: 1 | Cascade: 1

```
B G R B Y G B G R B Y
Y R Y Y R G B G G B Y
R B Y B G B Y B R G B
Y Y G G R Y R Y R B G
R R Y R R G R R G R B
R G R B B G G R B Y B
B B Y G Y R R Y G B Y
G Y G R R B Y B Y Y B
Y B G R G B Y R B R G
B B Y G G R R Y Y G B
B R Y G Y G Y R B B R
```

```

if self.scor >= SCOR_TINTA:
    self.joc_terminat = True
    self.motiv_oprire = "ATINS_SCOR_TINTA"
    return False
potential = []
for r in range(self.randuri):
    for c in range(self.coioane):
        if c+1 < self.coioane:
            potential.append((r,c,r,c+1))
        if r+1 < self.randuri:
            potential.append((r,c,r+1,c))
valid = []
for r1, c1, r2, c2 in potential:
    self.tabla[r1][c1], self.tabla[r2][c2] = self.tabla[r2][c2], self.tabla[r1][c1]
    formatiuni, puncte = self.gaseste_formatiuni()
    if formatiuni:
        valid.append({'mutare':(r1,c1,r2,c2), 'puncte': puncte})
    self.tabla[r1][c1], self.tabla[r2][c2] = self.tabla[r2][c2], self.tabla[r1][c1]
if not valid:
    self.joc_terminat = True
    self.motiv_oprire = "NU_EXISTA_MUTARI"
    return False
best = max(valid, key=lambda x: x['puncte'])
r1, c1, r2, c2 = best['mutare']
self.incearca_swap(r1, c1, r2, c2)

```

```

def pas_auto_joc_vizual(joc, delay=0.01): 1 usage
    if joc.scor >= SCOR_TINTA:
        joc.joc_terminat = True
        joc.motiv_oprire = "ATINS_SCOR_TINTA"
        return False
    potential = []
    for r in range(joc.randuri):
        for c in range(joc.coioane):
            if c+1 < joc.coioane:
                potential.append((r,c,r,c+1))
            if r+1 < joc.randuri:
                potential.append((r,c,r+1,c))
    valid = []
    for r1,c1,r2,c2 in potential:
        joc.tabla[r1][c1], joc.tabla[r2][c2] = joc.tabla[r2][c2], joc.tabla[r1][c1]
        formatiuni, puncte = joc.gaseste_formatiuni()
        if formatiuni:
            valid.append({'mutare': (r1,c1,r2,c2), 'puncte': puncte})
        joc.tabla[r1][c1], joc.tabla[r2][c2] = joc.tabla[r2][c2], joc.tabla[r1][c1]
    if not valid:
        joc.joc_terminat = True
        joc.motiv_oprire = "NU_EXISTA_MUTARI"
        return False
    best = max(valid, key=lambda x: x['puncte'])
    r1,c1,r2,c2 = best['mutare']
    joc.incearca_swap(r1,c1,r2,c2)
    clear_console()
    print(f"Swap: ({r1},{c1}) <-> ({r2},{c2}) | Scor: {joc.scor} | Swap-uri: {joc.swapuri} | Cascade: {joc.total_cascade}")
    joc.afiseaza_tabla()
    time.sleep(delay)
    return True

```

AUTOMATIZAREA MUTARILOR

- CODUL VERIFICA TOATE MUTARILE POSIBILE SI ALEGE MUTAREA CARE GENEREAZA CELE MAI MULTE PUNCTE.
- FUNCTII PRINCIPALE: PAS_AUTO_JOC() SI PAS_AUTO_JOC_VIZUAL().
- INCLUDE VIZUALIZARE CU DELAY PENTRU A URMARI PROCESUL DE JOC.
- DACA NU MAI EXISTA MUTARI VALIDE, JOCUL SE OPRESTE AUTOMAT.

CONCLUZII SI INCHEIERE

- AUTOMATIZAREA AJUTA LA TESTAREA STRATEGIILOR SI OPTIMIZAREA SCORULUI IN CANDY CRUSH.

POSIBILITATI DE IMBUNATATIRE:

- INTERFATA GRAFICA PENTRU VIZUALIZARE MAI ATRACTIVA.
- ALGORITMI PENTRU MUTARI OPTIME.

```
2 0,10000,127,414,True,ATINS_SCOR_TINTA,127
3 1,10000,151,450,True,ATINS_SCOR_TINTA,151
4 2,10000,174,481,True,ATINS_SCOR_TINTA,174
5 3,10000,133,397,True,ATINS_SCOR_TINTA,133
6 4,10000,143,422,True,ATINS_SCOR_TINTA,143
7 5,10000,162,434,True,ATINS_SCOR_TINTA,162
8 6,10000,126,367,True,ATINS_SCOR_TINTA,126
9 7,10000,166,459,True,ATINS_SCOR_TINTA,166
10 8,10000,127,385,True,ATINS_SCOR_TINTA,127
11 9,10000,171,507,True,ATINS_SCOR_TINTA,171
12 10,10000,147,416,True,ATINS_SCOR_TINTA,147
13 11,10000,154,442,True,ATINS_SCOR_TINTA,154
14 12,10000,123,393,True,ATINS_SCOR_TINTA,123
15 13,10000,161,460,True,ATINS_SCOR_TINTA,161
16 14,10000,117,362,True,ATINS_SCOR_TINTA,117
17 15,10000,119,392,True,ATINS_SCOR_TINTA,119
18 16,10000,125,377,True,ATINS_SCOR_TINTA,125
19 17,10000,124,404,True,ATINS_SCOR_TINTA,124
```

```
R G G B Y G R B Y R B
G R B G G R B G B Y G
Y R Y G Y Y R Y G B Y
B Y R Y B B Y Y R R G
B R G B G R B R B B Y
Y G R B B Y B B G Y R
```

Rezultatele au fost salvate in results/summary.csv