

PROG1: Projekt

Dolezite informacie:

- Projekt je za **10** bodov.
- Termin odovzdania projektu: **do nedele 10.12. 23:59**
- Pozorne si precitajte cast "**Odovzdavanie**" na konci zadania projektu.
- Projekty budete odovzdavat do miesta odovzdania v AISe.
- Pri odovzdavani **POZORNE** sledujte vypisy na obrazovke! Odovzdanie bude treba **VIACKRAT** potvrdit!
- Pozorne si precitajte cast "**Plagiatorstvo netolerujeme!**".

Plagiatorstvo netolerujeme!

- Prosim, odovzdajte vas **VLASTNY** zdrojovy kod.
- **NEPOSKYTUJTE** vas kod nikomu inemu. Ohrozujete tym samych seba!
- Vsetky odovzdane projekty dame do **ANTIPLAGIATORSKEHO** softveru, ktorý zisti zhody medzi projektami. Projekty porovname aj s projektami z minulých rokov.
- Antiplagiatorsky softver, ktorý používame je veľmi sofistikovaný. Urcite vám neprejde, ak odovzdáte okopirovaný kód, v ktorom budú zmenené mená funkcií, zmenené názvy premenných alebo preusporiadané bloky kódu. Nami používaný antiplagiatorsky softver taketo veci spoľahlivo odhalí.
- Neskusajte na internete hľadať riešenie projektu. Kód okopirovaný z internetu je stále okopirovaný kód! Minulý rok niektorí študenti našli na internete riešenie projektu a časti z tohto riešenia okopirovali. Antiplagiatorsky softver ich, samozrejme, odhalil. Mimochodom, riešenie, ktoré našli na internete, bolo veľmi nesikovne napísané a v prípade, že použijete niečo z toho riešenia, budeme to vedieť veľmi ľahko rozoznať.
- Okrem okopirovania projektu, sme si, samozrejme, vedomí aj možnosti, že študent dá projekt vypracovať niekomu inému. V prípade takehoto podozrenia budeme vyžadovať, aby študent pred nami svoj projekt odprezentoval!
- Vynatok zo študijného poriadku:
“Preukázaná nečestnosť študenta pri hodnotení študijných výsledkov (zistenie opisovania, podvádzanie, použitie nedovolených pomôcok a iných praktík vrátane nedovolenej spolupráce počas písomného alebo ústneho overovania vedomostí študenta, plagiátorstvo a pod.) má za následok neúspešné absolvovanie predmetu (čl. 16 tohto študijného poriadku). Takéto konanie študenta je porušenie zásad študijnej morálky a môže byť predmetom disciplinárneho konania.”

Zadanie projektu:

Cielom zadania je nasimulovať 2 hráčov hrajúcich *Clovece, nehnevaj sa*.

Pravidla hry *Clovece, nehnevaj sa* môžete nájsť na:

https://sk.wikipedia.org/wiki/%C4%8Clove%C4%8De_nehnevaj_sa

Pravidla hry v anglickom jazyku nájdete na:

https://en.wikipedia.org/wiki/Mensch_%C3%A4rgere_Dich_nicht

Pravidla hry nájdete vysvetlené aj v tomto videu:

<https://www.youtube.com/watch?v=IHkRjn8XVtw>

Je viacero verzii pravidiel. Ak sa trochu odchýlite od pravidiel na wikipédii alebo od pravidiel vo videu, nie je to problém.

Cast prva (2 body)

Naprogramujte funkciu *gensachovnicu* s parametrom n , ktorá vygeneruje a vráti "sachovnicu" rozmeru $n \times n$ pre *Clovece, nehnevaj sa*. (Budete sa musiet zamysliet nad tým, ako sachovnicu vhodne reprezentovat.) Dalej naprogramujte funkciu *tlacsachovnicu*, ktorá bude mat ako argument sachovnicu a ktorá tuto sachovnicu vykresli.

Sachovnica ma tvar kriza s ramenami sirky 3. Cislo n (t.j. rozmer sachovnice) musi byt neparne. Napriklad pre $n = 9$ vyzera sachovnica nasledovne:

```
      0 1 2 3 4 5 6 7 8
0      * * *
1      * D *
2      * D *
3 * * * * D * * *
4 * D D D X D D D *
5 * * * * D * * *
6      * D *
7      * D *
8      * * *
```

Pre $n=13$ vyzera sachovnica nasledovne:

```
      0 1 2 3 4 5 6 7 8 9 0 1 2
0      * * *
1      * D *
2      * D *
3      * D *
4      * D *
5 * * * * * * D * * * * *
6 * D D D D D X D D D D D *
7 * * * * * * D * * * * *
8      * D *
9      * D *
0      * D *
1      * D *
2      * * *
```

Cast druha (5 bodov)

Nasimulujte pohyb jedneho panacika po sachovnici. Pouzivatel si na zaciatku bude moct zvolit velkost sachovnice. Spociatku bude panacik umiestneny na rohu horneho ramena ako na obrazku:

```
    0 1 2 3 4 5 6 7 8
0      * * A
1      * D *
2      * D *
3 * * * * D * * *
4 * D D D X D D D *
5 * * * * D * * *
6      * D *
7      * D *
8      * * *
```

Nasledne sa vygeneruje nahodne cislo od 1 do 6. Na generovanie cisla pouzite funkciu `randint` z kniznice `random` (<https://docs.python.org/3.8/library/random.html>). Podla toho, ake cislo sa vygenerovalo, sa panacik posunie o dany pocet pozicii. Nasledne sa vygeneruje nove nahodne cislo a tak dalej. Simulacia konci, ked sa panacik dostane do domceka (na jednu z pozicii oznacenyh pismenom D) v hornom ramene. Simulacia musi vypisat kazde vygenerovane cislo a po vygenerovani noveho cisla musi simulacia vzdy vykreslit sachovnicu s aktualnou poziciou panacika.

Cast tretia (3 body)

Nasimulujte hru 2 hracov A,B. Pouzivatel si na zaciatku bude moct zvolit velkost sachovnice. Hrac A startuje na pozicii v rohu horneho ramena, Hrac B startuje na pozicii v rohu spodneho ramena (vid obrazok nizsie).

```
    0 1 2 3 4 5 6 7 8
0      * * A
1      * D *
2      * D *
3 * * * * D * * *
4 * D D D X D D D *
5 * * * * D * * *
6      * D *
7      * D *
8      B * *
```

Kazdy hrac ma $(n-3)/2$ panacikov. Vyhra ten hrac, ktory ako prvý umiestni vsetky svoje figurky do domceka. Hrac A sa snazi dostat figurky do domceka v hornom ramene, hrac B sa snazi dostat figurky do domceka v spodnom ramene. Hadzanie kocky opat simulujte pomocou generovania nahodnych cisel. Vasa simulacia musi vypisat kazde vygenerovane cislo a po vygenerovani noveho cisla musi simulacia vzdy vykreslit sachovnicu s aktualnou poziciou panacikov.

Vasa simulacia moze fungovat napríklad tak, ze hrac sa bude hybat s jednou figurkou, az kym ju nedostane do domceka, a az potom nastavi na sachovnicu dalsiu figurku. To znamena, ze kazdy hrac bude mat v kazdom okamihu hry na sachovnici najviac jednu figurku.

Vasa simulacia ale moze fungovat aj tak, ze hrac bude moct mat na sachovnici viacero figurok

sucasne. V takom pripade si mozete vybrat, podla akeho pravidla sa bude hrac rozhodovat, ktorou figurkou sa pohne. (Napriklad sa hrac moze pohnut vzdy tou figurkou, ktora je najblizsie pri domceku.)

Poznamka:

V tejto ulohy mate vytvorit program, ktorý používateľovi umožní spustiť **simuláciu** hry *Clovece, nehnevaj sa* medzi dvoma hráčmi. Používateľ by si mal na začiatku zvoliť veľkosť šachovnice a následne by sa mala spustiť simulácia. Simulácia by mala skončiť, až keď jeden hráč vyhraje, a nemala by od používateľa vyžadovať žiadne ďalšie vstupy. Po spustení simulácie teda používateľ uvidí v konzole veľa výpisov a veľa šachovnic a nakoniec sa dozvie, ktorý hráč vyhral bez toho, aby používateľ do hry nejak zasahoval.

Ak chcete, môžete ale namiesto takejto simulácie vytvorit program, ktorý používateľovi umožní "zahrať" si hru *Clovece, nehnevaj sa* pre dvoch hráčov. Takyto program by používateľa vyzýval, nech zvolí, ktorou figurkou sa má hráč pohnúť.

Odovzdávanie:

- Do miesta odovzdania v AISE nahrajte **jeden** skript s celým vaším zdrojovým kódom.
- Súčasťou hodnotenia bude aj formálna stránka kódu. Váš kód by mal byť členený do funkcií a vhodne okomentovaný (inak mínusové body).
- Po spustení skriptu by sa mala spustiť najpokročilejšia simulácia, ktorú ste dokázali urobiť.
Čize:
 - buď sa iba vykreslí šachovnica (alebo nejaka jej nedokonala verzia),
 - alebo sa spustí simulácia pohybu jedného panáčka (alebo nejaka jej nedokonala verzia),
 - alebo sa spustí simulácia celej hry *Clovece, nehnevaj sa* pre dvoch hráčov (alebo nejaka jej nedokonala verzia).
- Na začiatku skriptu uveďte komentár, v ktorom popíšete:
 - ktoré časti projektu váš skript rieši
 - ak skript rieši niektoré časti nedokonalé, popíšete, o ake nedokonalosti sa jedná
 - ľubovoľne ďalšie informácie, na ktoré chcete hodnotiaceho upozorniť
- Pri odovzdávaní nezabudnite čo? ... **POZORNE SLEDOVAŤ VYPISY NA OBRAZOVKE !**
- Odovzdanie bude treba čo? ... **VIACKRÁT POTVRDIŤ !**

ENJOY! :)