

## Wstęp do programowania

### Pracownia 13

**Uwaga:** Wszystkie zadania z tej listy będzie można oddawać do końca semestru. Wszystkie zadania są z gwiazdką (czyli wartość maksimum nie jest modyfikowana). Za tę listę nie ma premii.

Po kolokwium pojawi się jeszcze minilista 14 z zadaniami z tego kolokwium (można liczyć na około 2 punkty)

**Zadanie 1.(1pkt)(\*)** Rozwiąż zadanie z zeszłych list, jakiego jeszcze nie robiłeś (lub wykonaj operację upgrade na zadaniu za 0.5) lub rozwiąż zadanie z Analizy Literackiej, którego jeszcze nie robiłeś.

**Zadanie 2.(1pkt)** Program labirynt.py z Wykładu 13 jest trochę zbyt długi. Podziel go na 4 pliki (usuając przy okazji niepotrzebne fragmenty):

- a) Plik zawierający całą część związaną z grafiką żółwia
- b) Plik zawierający przeszukiwanie wstecz
- c) Plik zawierający generację labiryntów
- d) Właściwy program

Zastanów się, jak to zrobić najlepiej. Być może będziesz musiał zmodyfikować interfejs pewnych funkcji, żeby poza głównym programem nie występowały żadne dane globalne.

**Zadanie 3.(0.5pkt)** Zaproponuj inny sposób generowania labiryntów, które są ciekawsze od prezentowanego na wykładzie. Możesz skorzystać z jakiegoś podzbioru poniższych pomysłów, lub zaproponować coś własnego:

1. Zamienić tworzenie murków na drażnienie korytarzy
2. Drażnić korytarze o współrzędnych nieparzystych (tzn. pionowe o nieparzystych x-ach i poziome o nieparzystych igrekach)
3. Zaimplementować „błądzenie”, czyli rysowanie długiej kreski z kwadratów, która skręca w losowych momentach.

Zaprezentuj działanie na programie, który koloruje komnaty.

**Zadanie 4.(1.0pkt)** Zmodyfikuj generowanie labiryntów o dodatkowy etap, w którym staramy się łączyć komnaty w jeden spójny system wykopując możliwie jak najmniej dodatkowych korytarzy (czyli zamieniając minimalną liczbę czarnych pól na białe). Program może wykopać co najwyżej K dodatkowych korytarzy. Wszystkie komnaty, których nie da się połączyć z główną (największą) komnatą powinny być zaczerwienione.

**Zadanie 5.(2pkt)** Rozwiąż zadanie o listach z dowiązaniem ze SKOS-u.

**Zadanie 6.(0.2+Xpkt)** Wypełnij zgłoszenie konkursowe Konkursu Alfabetonicznego

**Zadanie 7.(1+Xpkt)** Załóż profil na CodingGames i napisz w Pythonie bota do gry Fantastic Bits ( <https://www.codingame.com/multiplayer/bot-programming/fantastic-bits>). 1 punkt można otrzymać za program, który albo dostanie się do wyższej ligi drewnianej, albo zostanie uznany przez prowadzącego za nietrywialny. Oprócz tego z zadaniem związane są premie (sumujące się):

- 0.5 za dojdzie do ligi brązowej
- 0.5 za dojdzie do ligi srebrnej
- 1.0 za dojdzie do ligi złotej
- 1.0 za poziom Legendary
- Y za napisanie najlepszego bota wśród wszystkich uczestników kursu WdP. Wartość Y zależy od liczby uczestników tego minikonkursu, jest równa  $\min(1, 0.1 * (\text{liczba\_uczestnikow}-1))$