

Wstęp do programowania

Lista 11

Uwaga: Premia za tę listę wynosi 0.5, przyznawana jest osobom, które zdobyły co najmniej 2p za zadania z tej listy. Maksimum dla tej listy wynosi 3p. Zadania z tej listy można oddawać za 100% na zajęciach w tygodniu rozpoczynającym się 20 stycznia 2025 (ale jedynie za 50% na dodatkowych, zapowiedzianych zajęciach na początku sesji egzaminacyjnej).

Zadanie 1.(★pkt) Prezent Noworoczny: rozwiąż zadanie, którego do tej pory nie robiłeś.

Zadanie 2.(1pkt) Zdefiniujmy następujące przekształcenie na słowach (nazwiemy je *permutacyjną postacią normalną*): zamieniamy litery na liczby, w ten sposób, że:

1. Tym samym literom przypisane są równe liczby, różnym literom – różne liczby.
2. Liczby przypisywane są po kolei, licząc od lewej strony.

Otrzymane liczby sklejamy w jeden napis, wstawiając na przykład znak "-" jako separator. Przykładowe pary słowo i wartość przekształcenia: tak: 1-2-3, nie: 1-2-3, tata: 1-2-1-2, indianin: 1-2-3-1-4-2-1-2. Napisz funkcję, która zwraca w wyniku wartość opisanego przekształcenia. *To zadanie będzie miało kontynuację na kolejnej liście.*

Zadanie 3.(1pkt) Twoim zadaniem będzie zasymulowanie wyborów parlamentarnych w Polsce, opierając się na rzeczywistych wynikach z poszczególnych okręgów. Na stronie wykładu będą wyniki z przedostatnich wyborów, możesz je zamieniać na wyniki ostatnich, ale sam musisz przystosować dane PKW do zadania. Od jakiegoś czasu w Polsce obowiązuje metoda D'Hondta przydzielania mandatów. Poniżej skrócony opis tej metody (w razie potrzeby możesz zajrzeć na przykład do Wikipedii, gdzie jest ona dokładniej opisana).

- Każdy komitet ma wynik (będący liczbą rzeczywistą), rozdzielamy M mandatów pomiędzy N komitetów, zależnie od wyniku danego komitetu.
- Tworzymy ilorazy tych wyników przez kolejne liczby całkowite od 1 do M (dla każdego ilorazu zapamiętujemy, z jakiego komitetu pochodzi).
- Sortujemy te $M \times N$ liczb malejąco. Bierzymy pierwsze M i przydzielamy odpowiednim komitetom M mandatów.

Pomiń kolumnę Inne (to komitety, które nie przekroczyły progu wyborczego). Oblicz sumaryczną liczbę mandatów dla każdego ugrupowania. Sprawdź, czy otrzymałeś te same wyniki, co Państwowa Komisja Wyborcza.

Zadanie 4.(1pkt)(★) Zadanie to wymaga rozwiązania zadania o relacjach równoważności z poprzedniej listy (możesz je rozwiązać również teraz za pełną liczbę punktów), odwołujemy się tu tegorocznych wyborów parlamentarnych. Załóżmy, na potrzeby tego zadania, że „elektoraty się dodają”, to znaczy, że jeżeli (przykładowo) Nowa Lewica oraz Konfederacja Wolność i Niepodległość zdecydowałyby się na wspólny start, to na taką połączoną listę zagłosowałyby dokładnie te same osoby, co w ostatnich wyborach na pojedyncze komitety. Napisz program, który przeglądając wszystkie relacje równoważności na komitetach wyborczych (możesz skoncentrować się na pięciu, które weszły do sejmu) przeprowadza symulowane wybory dla różnych sojuszy (czyli dla różnych decyzji o połączeniu komitetów). Twój program powinien:

- a) Wypisać wszystkie warianty wyborów, w których istnieje koalicyjny komitet posiadający większość mandatów i niezawierający ugrupowania, które zdobyło najwięcej głosów (czyli PiS).
- b) Wypisać wszystkie warianty wyborów, w których istnieje koalicyjny komitet posiadający większość konstytucyjną.
- c) Wypisując wariant wyborów wypisz sojusze i liczbę mandatów dla poszczególnych sojuszy

Zadanie 5.(1pkt) Popraw klasę `Set` z wykładu 12, by obsługiwała również takie konstrukcje jak `len(s)` oraz `s1 & s2`, `s1 - s2`

(uwaga: częścią tego zadania jest samodzielne znalezienie, jakie nazwy muszą mieć `__specjalne_metody__` (*dunder methods*) przeciążające operatory.

Zadanie 6.(0.5+1pkt) Przygotuj własną animację, korzystając z biblioteki `turtle`¹ (możesz w dowolny sposób korzystać z kodu z wykładu 11). Jedynymi „sztywnymi” wymaganiami jest zachowanie płynności animacji i dodanie zmieniających się płynnie kolorów pewnych elementów animacji, reszta zależy od Twojej inwencji (w szczególności możesz zmienić również tematykę animacji). Jeżeli zachwycisz prowadzącego, może dać Ci uznaniową premię w wysokości do 1p.

Zadanie 7.(0.5+2pkt) Dodaj do animacji element sterowania i (opcjonalnie) jakiś minimalny aspekt grywalności. Na wykładzie 16.01 pojawią się przykłady takich programów używających biblioteki `turtle`. Jeżeli znasz bibliotekę `pygame` (której nie będzie na naszym wykładzie), możesz skorzystać z niej zamiast z `turtle`.

¹Lub innej, umożliwiającej wykonywanie rysunków zbudowanych z linii, wielokątów, etc