

**Programowanie Obiektowe**  
**Podstawy języka Java**  
**Zadanie oceniane nr 1b**  
**1-04-2021**

Po zakończeniu pracy konieczne jest wgranie zmian do repozytorium (add + commit + push) w katalogu o nazwie w stylu: zadanie\_oceniane\_1. Fakt wgrania plików do swojego repozytorium można sprawdzić samodzielnie logując się (via www) na swoje konto i sprawdzając czy pojawiły się tam wszystkie zmiany.

## "Górnika"



W dniu dzisiejszym należy stworzyć mini symulator górnika, pracującego pod ziemią. Terenem jego działania jest struktura którą można wyobrazić sobie taką, jaka jest zaprezentowana na załączonym obrazku. Szybem jest tablica referencji na inne tablice będące korytarzami. Korytarze "nie odchodzą" od każdej komórki szybu, lecz mają swoją referencję w co któreś z nich. Pozostałe komórki są puste (null).

|      |           |  |          |          |           |          |  |           |           |
|------|-----------|--|----------|----------|-----------|----------|--|-----------|-----------|
| null |           |  |          |          |           |          |  |           |           |
| ->   |           |  |          |          |           |          |  |           |           |
| null |           |  |          |          |           |          |  |           |           |
| ->   |           |  |          |          | chalkolit |          |  |           |           |
| ->   |           |  |          |          |           |          |  | chalkolit |           |
| null |           |  |          |          |           |          |  |           |           |
| null |           |  |          |          |           |          |  |           |           |
| ->   |           |  |          | uraninit |           |          |  |           |           |
| null |           |  |          |          |           |          |  |           |           |
| null |           |  |          |          |           |          |  |           |           |
| ->   |           |  | uraninit |          |           |          |  |           |           |
| ->   |           |  |          |          |           | uraninit |  |           |           |
| ->   | chalkolit |  |          |          |           |          |  |           | chalkolit |
| null |           |  |          |          |           |          |  |           |           |

## Prace do wykonania:

1. Stworzyć hierarchię klas odzwierciedlającą elementy występujące w tym zadaniu (nazwijmy je Obiektami Kopalnianymi) wraz z danymi które przechowują (podane w nawiasach):

- uraninit (forma\*; radioaktywność (losowana z przedziału 700-950); bazowa wartość rynkowa)
- skała (bazowa wartość rynkowa – ustawiana na 1)
- chalkolit (radioaktywność – losowana z przedziału 700-950; bazowa wartość rynkowa)

*\* uranofan albo torbernit albo fomalinit – wylosowana z jednakowym prawdopodobieństwem jedna z tych trzech możliwych wartości*

Bazowa wartość rynkowa jest typem pozycyjnym, radioaktywność całkowitą. Konkretnie typy dobierać wedle uznania. Pola bez wytycznych co do ich tworzenia są przekazywane z zewnątrz, pozostałe kreowane są "na terenie" danej klasy w momencie jej instancjonowania.

2. Stworzyć klasę Adit zarządzającą modelem kopalni. Powinna ona:

- podczas jej instancjonowania (w oparciu o przekazane parametry zawierające informacje: maksymalny dystans pomiędzy korytarzami, długość korytarzy, ilość korytarzy) utworzyć dwuwymiarową tablicę reprezentującą szyb wraz z korytarzami. Szyb (kolory niebieskie na obrazku) jest tablicą referencji korytarze (kolor szary na obrazku) będące tablicami przechowującymi referencje na Obiekty Kopalniane. Korytarze nie odchodzą od każdej komórki szybu, tylko od co którejś. Długość szybu

ma wynosić iloczyn maksymalnego dystansu pomiędzy korytarzami oraz ich ilości zwiększonej o 1. W momencie inicjalizacji tej struktury, zanim każdy następny korytarz zostanie wstawiony jako odnoga od szybu, należy wylosować wartość (z przedziału 1–maksymalny dystans pomiędzy korytarzami) mówiącą o ile komórek dalej (od bieżącego korytarza) będzie on (ten kolejny) umiejscowiony. Chodzi o to, żeby korytarze pojawiały się nieregularnie. Wypełniamy je losowo obiektami reprezentującymi uraninit, chalkolit. Prawdopodobieństwo tego, że napotkana komórka korytarza wypełniona będzie uraninitem wynosi 0.1, chalkolitem 0.2, a skałą 0.7. Bazowa wartość rynkowa przekazana dla obu minerałów podczas ich instancjonowania to 50.

- Posiadać metodę wypisującą strukturę kopalni na konsolę wedle następujących reguł: pusta (null) komórka szybu – spacja, komórka szybu od której odchodzi korytarz – "->", pusta komórka korytarza (null), wartość rynkowa niepustej komórki korytarza (jako jej **reprezentacja tekstowa**). Metoda używana tylko na terenie klasy Adit.

### 3. Stworzyć klasę reprezentującą Górnika i wkomponować ją w klasę Adit. Zmodyfikować hierarchię Obiektów Kopalnianych).

Górnik (income = 0, odnośnik do struktury kopalni, przekazywany w momencie tworzenia)

- Stworzyć interfejs MineryObjectEvaluator implementowany przez główną nadklasę stworzonej hierarchii Obiektów Kopalnianych zawierający metodę domyślną double getModifiedMarketValue() (zwracająca 0.0). Dla chalkolinitu metoda ta zwraca jego bazową wartość handlową. Dla uraninitu również będzie to bazowa wartość handlowa pomnożona przez współczynnik ściśle związany z formą (i tylko z nią) pod którą uraninit występuje: uranofan x 1.0, torbernit x 1.5 i fomarieryt x 2.5.
- Dodać do klasy Adit pole będące odnośnikiem do typu reprezentującego górnika oraz umożliwić jego stworzenie i ustawienie za pomocą istniejącego konstruktora
- Zaimplementować w klasie reprezentującej górnika metodę goMiner(), w której:
  - ✓ przechodzi on przez wszystkie korytarze w kopalni
  - ✓ na każdej niepustej komórce wywołuje metodę: getModifiedMarketValue() i dodaje zwrócony rezultat do pola income
  - ✓ wstawia null w każdą odwiedzoną komórkę