

Ćwiczenia Laboratoryjne

Zasady wykonywania ćwiczeń

Ćwiczenia laboratoryjne wykonywane są na komputerach w laboratorium, bez możliwości skorzystania z internetu. Możliwe jest używanie własnych notatek oraz korzystanie z dokumentacji bibliotek Java.

Każde ćwiczenie przygotowane jest w formie kodu, który można zaimportować do NetBeans jako projekt z istniejącymi źródłami. W projekcie istnieją jedynie testy umożliwiające sprawdzenie, czy napisany kod jest prawidłowy i można automatycznie sprawdzić swoją ocenę (pięć scenariuszy testowych, każdy za 1 punkt). Wszystkie testy jednostkowe muszą wykonać się prawidłowo, aby zaliczyć dany scenariusz testowy.

Zadania można wykonywać w dowolnej kolejności, jednakże muszą one być wykonywane precyzyjnie, zgodnie z poleceniem (np. nazwy klas muszą być takie same jak w poleceniu, pakiety, klasy itd. również muszą być takie, jak wskazane w poleceniu).

W wyjątkowych sytuacjach lub w przypadku znalezienia błędu w testach (kod jest napisany ewidentnie prawidłowo, a testy nie przechodzą) należy zgłosić to prowadzącemu i w przypadku potwierdzenia błędu w przygotowanym ćwiczeniu, punkty zostaną przyznane "ręcznie".

Oprócz kodu pisanego w celu wykonania ćwiczenia, można pisać dowolne inne metody (np. funkcję main), uruchamiać dowolny kod, tak aby spełnić przygotowane testy. Nie wolno zmieniać testów i w wymyślny sposób pisać kod tak, aby ominąć ich wykonanie (do oceny przez prowadzącego).

W trakcie ćwiczeń można zadawać pytania prowadzącemu (w granicach przyzwoitości). Nie wolno korzystać z pomocy innych osób i w jakikolwiek sposób próbować oszukać prowadzącego zajęcia. Nieuczciwość będzie skutkowała zakończeniem uczestnictwa w zajęciach do końca semestru i ukończenie ich z oceną niedostateczną, bez możliwości poprawy. Proszę pamiętać, że korzystanie z cudzego konta na moodle jest formą oszukiwania i każdy taki przypadek można łatwo wykryć analizując logi systemowe.

Lokalizacja plików

Zarówno treść zadania, jak i projekt będą przez odpowiednią chwilę dostępne na platformie **e.uksw.edu.pl**. Należy ściągnąć plik z projektem oraz treść polecenia. Brak dostępu do tych plików należy niezwłocznie zgłosić prowadzącemu, który może w wyjątkowych sytuacjach przegrać treść i pliki z innych źródeł danych (np. pendrive).

Utworzenie projektu

W przypadku, gdy zwykle otwarcie projektu nie powiedzie się z błędem "reference problems", to należy zignorować błąd i wybrać odpowiednią platformę wykonując odpowiednie kroki

1. Wejść we właściwości projektu (prawy klawisz myszy na projekcie na liście projektów i opcja "properties")
2. Przejść do zakładki "Libraries"
3. Zmienić platformę na domyślną (Default)

W przypadku, gdy zwykle otwarcie projektu nie powiedzie się, należy utworzyć projekt z istniejących źródeł. Należy wykonać następujące kroki:

1. Z folderu z projektem usunąć wszystkie pliki za wyjątkiem folderów `src` i `test`
2. Utwórz nowy projekt (**ctrl+shift+n**)
3. Wybierz projekt Java -> Java project with existing sources
4. Nadaj nazwę projektowi zgodnie z numerem ćwiczenia (PRJA1, PRJA2, itd.)

5. Wskaż (Browse...) folder na pulpicie zawierający projekt testowy i należy przejść do następnego ekranu kreatora
6. Na kolejnym ekranie należy wskazać foldery z plikami źródłowymi (`src`) oraz testami (`test`)
7. Na kolejnym ekranie można nacisnąć next i zakończyć kreatora.

Sprawdzenie projektu

Aby sprawdzić swoją punktację (tak będą oceniane projekty) można wykonać test projektu (skrót klawiszowy `alt+F6`).

Oddanie projektu

Aby oddać projekt do oceny prowadzącemu, należy wysłać spakowany ZIPem projekt na platformę `e.uksw.edu.pl` pod odpowiednie ćwiczenie. Wysłanie zadania należy zatwierdzić wybierając opcję wyślij do oceny. Ćwiczenia niezatwierdzone nie będą sprawdzane.

Ocena

Na podstawie testów automatycznych, ew. uwag zgłoszonych podczas ćwiczeń przyznane zostaną punkty, które pojawią się na platformie moodle. Ocena za ćwiczenie powinna pojawić się przed kolejnymi zajęciami.

Powodzenia!

Laboratorium 2

Wszystkie zadania należy wykonać w klasie `Automat` pakietu `pl.edu.uksw.wmp.prja.laboratorium2` (od jej utworzenia należy rozpocząć ćwiczenie). Klasa ta ma w założeniu reprezentować automat do kawy, podobny do tych, które stoją na korytarzach UKSW. Oczywiście, odwzorowujemy jedynie logikę jego działania.

Zadanie 1

Do klasy `Automat` dodaj prywatne atrybuty `iloscKubkow` oraz `iloscMleka` oznaczające odpowiednio ilość kubków, które pozostały w automacie oraz ilość mleka, które można dodać do kawy. Mleko powinno być typu `double`, a ilość kubków typu `int`.

Zadanie 2

Do klasy `Automat` dodaj publiczny konstruktor domyślny, inicjalizujący ilość kubków na 10 oraz ilość mleka na 100 oraz publiczny konstruktor argumentowy inicjalizujący oba atrybuty (nazwy argumentów identyczne z nazwami atrybutów).

Zadanie 3

Do klasy `Automat` dodaj metody publiczne dodające kubki i mleko (`dodajKubki`, `dodajMleko`). Należy pamiętać, że w automacie nie może być więcej niż 1000 kubków oraz 200 jednostek mleka. Próba dodania większej ilości powinna zakończyć się ustaleniem maksymalnej wartości danego atrybutu.

Zadanie 4

Do projektu dodaj typ wyliczeniowy o nazwie `RodzajKawy` przyjmujący jedną z czterech możliwych pozycji: `KAWA_Z_MLEKIEM`, `KAWA_CZARNA`, `KAWA_Z_MLEKIEM_I_CUKREM`, `KAWA_CZARNA_Z_CUKREM`. Do klasy `Automat` dodaj metodę `podajKawe`, której argumentem oraz typem zwracanym będzie `RodzajKawy`. Metoda ma działać według następującej logiki:

- Automat wydając kawę zużywa jeden kubek,
- Kawa z mlekiem zużywa 0.5 jednostki mleka, a kawa z mlekiem i cukrem 0.3 jednostki mleka
- W przypadku niemożliwości wydania kawy, należy zwrócić `null`

Zadanie 5

Zmodyfikuj klasę tak, aby obsługiwała płatność za kawę monetami, przy czym należy kontrolować ilość monet w automacie i możliwość wydania reszty. Należy dodać metodę `podajKawe` z argumentem typu `RodzajKawy` oraz drugim, będącym tablicą liczb typu `Moneta` (dodana do projektu, nie można zmieniać). Metoda ta powinna zwracać tablicę typu `Moneta`. Dodaj też metodę `opoznij` opróżniającą automat z wrzuconych monet (zostawiający po 5 monet każdego typu).

Należy przyjąć, że

- Każdy nowy automat ma po 5 monet każdego typu,
- Monet danego typu może być w automacie maksymalnie 50,
- Kawa z mlekiem kosztuje 1.80zł, a kawa czarna 1.60zł (niezależnie czy jest z cukrem, czy bez).
- Algorytm wydawania reszty może być dowolny, byle wydał on prawidłową wartość,
- W przypadku braku możliwości wydania reszty, metoda powinna zwrócić `null`,