

Задание. Выделить классы эквивалентности для учебной функции

Выполните анализ входных параметров и разбиение на подобласти (классы эквивалентности) для функции, которая по дате определяет день недели.

Сначала попробуйте это сделать на бумажке, пользуясь здравым смыслом и интернетом.

Затем попробуйте проверить, подойдут ли составленные наборы тестовых данных для тестирования учебного приложения (ссылка на него находится в самом конце задания).

После ознакомления с тестируемым приложением можете добавить ещё какие-то наборы тестовых данных к своему списку.

Затем выполните тесты для этого приложения, запишите результаты тестирования (какие тесты выполняли, какие баги нашли). В процессе тестирования тоже можно добавлять новые тестовые наборы.

В финальном отчёте отметьте, какие тестовые данные на каком этапе были созданы (до знакомства с реализацией, после знакомства с реализацией, в процессе выполнения тестов). Если не все тесты были придуманы на самом первом этапе - постарайтесь объяснить, почему так получилось.

Ссылка на учебное приложение (**не ходите по ссылке, пока не выполните первый этап задания!**)

Текст ответа

1. Разбиение входных данных на классы эквивалентности (сделано до запуска программы):

Входные данные: конкретные вводимые даты и то, каким именно образом они вводятся

Дата:

Нельзя ≤ 0

Всегда можно ≥ 1 , но ≤ 28

Особые точки 29, 30, 31 (проверять 29 февраля в високосных годах, чередование 30 и 31 в разных годах)

Месяц:

Нельзя ≤ 0

Всегда можно ≥ 1 , но ≤ 12

Год:

Нельзя ≤ 0

Если не указаны никакие ограничения:

Високосные годы: хорошо бы здесь автоматизировать алгоритм проверки, так как сложно вручную проверить даже сто лет до сегодняшней даты и сто лет вперед. Это пятьдесят тестов.

Алгоритм: Год високосный, если он делится на четыре без остатка, но если он делится на 100 без остатка, это не високосный год. Однако, если он делится без остатка на и на 4, и на 100, и на 400, это високосный год.

Вручную проверить так: 1600, 2000, 2400 – так как выполняются все три условия (заодно сразу проверим макс и мин значения, так как уже могут попасть в эти области)

Максимальное ограничение можно установить произвольно. При тестировании проверять сотнями - программа должна выдать подсказку, какое минимальное и максимальное значения года может быть.

2. Предполагаемые тесты (написано до запуска программы):

Все числа – целые

Римские или арабские

Месяц – числом или буквами (возможно латинскими)

Ввод вручную или выбором из календаря, или копированием-вставкой

Порядок ввода: день, месяц, год или в другом порядке

Должны быть маски или подсказки типа: ДД.ММ.ГГГГ

Сегодняшнюю дату можно проверить, чтобы убедиться, что календарь настроен правильно. Ну и вчера, и завтра. Дополнительно можно неделю вперед и неделю назад.

Проверить все даты по классам эквивалентности (правильно ли указаны месяцы, где 30 дней, где 31, 28 в феврале, високосные годы, граничные значения)

3. Дополнительные наборы тестовых данных (после первого ознакомления с программой)

В поля ввода вписать другие символы - не числовые.

Баг - программа разрешает вводить буквы и не дает никаких подсказок. При выводе результатов сбрасывает на сегодняшнюю дату

Визуально просмотреть единообразие оформления информации

Баг - часть текста на английском языке, часть на русском

4. Проведенные тесты и баги:

Все числа – целые. Баг - можно вводить запятую, при этом год "сбрасывается" на год, который соответствует числам до запятой. Например, можно увидеть год - 0022 и выдается результат, никаких сообщений об ошибке

Римские или арабские. Программа допускает ввод букв, поэтому вписать римские возможно без предупреждений об ошибке

Месяц – числом или буквами (возможно латинскими). Есть подсказки и ограничения по количеству символов, поэтому можно догадаться, что только числами.

Ввод вручную или выбором из календаря, или копированием-вставкой. Получается и вручную, и копированием. Календаря нет.

Порядок ввода: день, месяц, год или в другом порядке. Есть подсказки, поэтому понятен порядок.

Должны быть маски или подсказки типа: ДД.ММ.ГГГГ. Масок нет, есть подсказки

Сегодняшнюю дату можно проверить, чтобы убедиться, что календарь настроен правильно. Ну и вчера, и завтра. Дополнительно можно неделю вперед и неделю назад. - Проверено, всё совпало.

Проверить все даты по классам эквивалентности (отрицательные, правильно ли указаны месяцы, где 30 дней, где 31, 28 в феврале, високосные годы, граничные значения). Проверено - указывается корректно. Если вводить несуществующую комбинацию, то перескакивает на день вперед. Если вводить отрицательные даты, то перескакивает на минус столько дней, месяц просто исправляет на положительный, а год так и оставляет с минусом. Предупреждений не выдает.

В поля ввода вписать другие символы - не числовые. Баг - программа разрешает вводить буквы и не дает никаких подсказок. При выводе результатов сбрасывает на сегодняшнюю дату

Визуально просмотреть единообразие оформления информации. Баг - часть текста на английском языке, часть на русском.

5. Итог.

Изначально при написании тестов представляла немного другую программу, поэтому некоторые тесты негодились. Например, Месяц – числом или буквами или выбор даты из календаря.

Критических багов не найдено. Достичь цели получилось при всех положительных тестах. Скорее, есть трудности с оформлением и удобством программы. И некорректные данные разрешает вводить и не выводит никаких предупреждающих сообщений - это плохо и неудобно.

ОТЗЫВ

Оценка	сдано!
Оценено в	Пятница, 4 февраля 2022, 15:34
Оценено	Homework Review
Отзыв в виде комментария	отлично