

Практическое занятие № 3

Вариант 10

Тема: составление программ ветвящейся структуры в IDE PyCharm Community.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ ветвящейся структуры в IDE PyCharm Community.

| Постановка задачи 1

Даны два целых числа: А, В. Проверить истинность высказывания: «Ровно одно из чисел А и В нечетное».

| Алгоритм

1. Ввод данных:

- Запросить у пользователя ввод первого целого числа А.
- Запросить у пользователя ввод второго целого числа В.

2. Проверка четности:

- Определить, является ли число А нечётным. ($isAodd = A \% 2 \neq 0$)

- Определить, является ли число В нечётным. ($\text{isBodd} = B \% 2 \neq 0$)

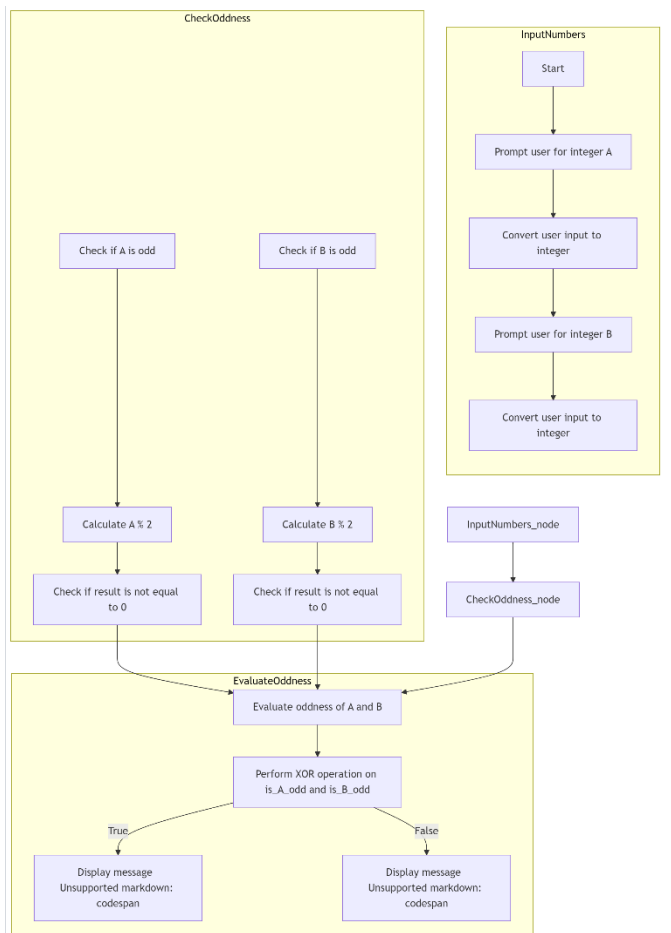
3. Проверка условия:

- Использовать логическую операцию "исключающее ИЛИ" для проверки, что ровно одно из чисел А или В нечётное.

- Если условие истинно, вывести сообщение: "Ровно одно из чисел А и В нечетное."

- В противном случае, вывести сообщение: "Либо оба числа четные, либо оба нечётные."

| Блок-схема



| Текст программы на Python

```
A = int(input("Введите целое число A: "))  
  
B = int(input("Введите целое число B: "))  
  
is_A_odd = A % 2 != 0  
  
is_B_odd = B % 2 != 0  
  
if is_A_odd ^ is_B_odd: # Исключающее ИЛИ  
    print("Ровно одно из чисел A и B нечетное.")  
else:  
    print("Либо оба числа четные, либо оба нечётные.")
```

| Протокол работы программы

1. Запустить программу в IDE PyCharm Community.
2. Ввести целое число A (например, 3).
3. Ввести целое число B (например, 4).
4. Программа проверяет, является ли A нечётным ($3 \% 2 \neq 0 = \text{True}$) и B нечётным ($4 \% 2 \neq 0 = \text{False}$).
5. Программа выполняет логику исключающего ИЛИ: $\text{True} \wedge \text{False} = \text{True}$.
6. Программа выводит сообщение: "Ровно одно из чисел A и B нечетное."

7. Если бы, например, А было 2, а В - 4, программа должна была бы вывести: "Либо оба числа четные, либо оба нечётные."

| Постановка задачи 2

Дан номер месяца — целое число в диапазоне 1-12 (1 — январь, 2 — февраль и т. д.). Определить количество дней в этом месяце для невисокосного года.

| Алгоритм

1. Ввод данных:

- Запросить у пользователя ввести номер месяца (целое число от 1 до 12).

2. Определение количества дней:

- Если номер месяца равен 1, 3, 5, 7, 8, 10 или 12, установить количество дней в 31.

- Если номер месяца равен 4, 6, 9 или 11, установить количество дней в 30.

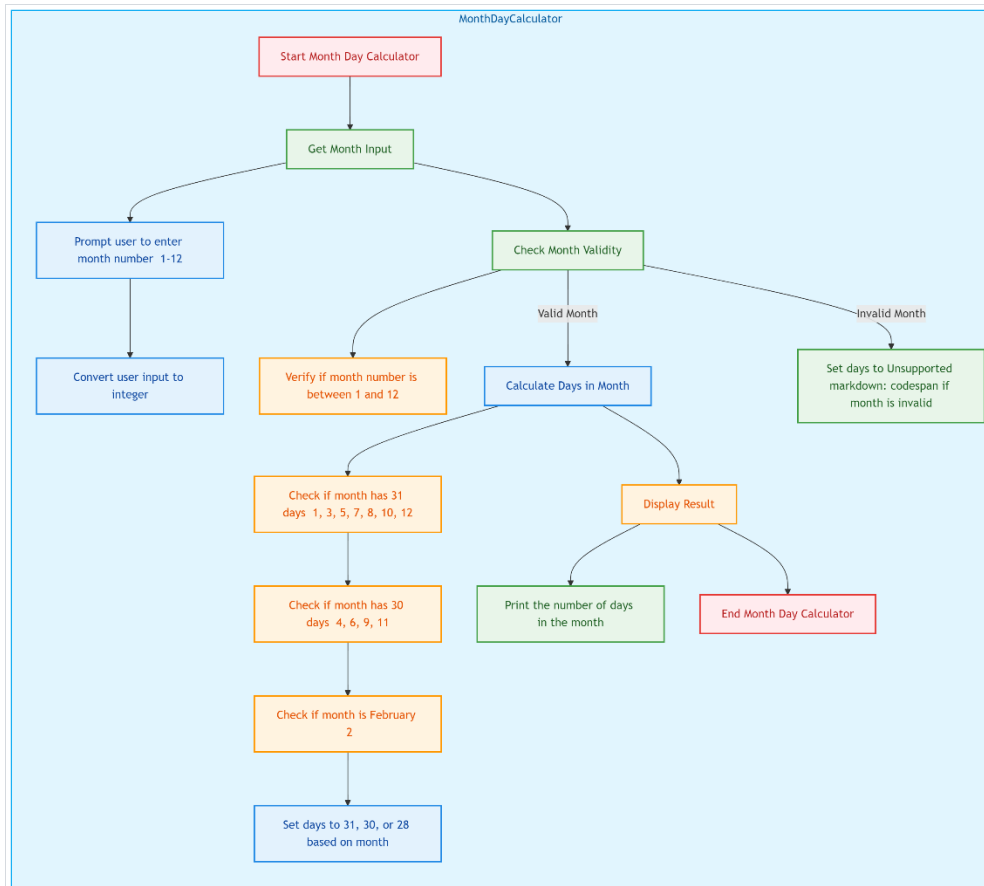
- Если номер месяца равен 2, установить количество дней в 28.

- Если номер месяца не соответствует диапазону от 1 до 12, вывести сообщение "Неверный номер месяца."

3. Вывод результата:

- Вывести количество дней в соответствующем месяце или сообщение об ошибке.

| Блок-схема



| Текст программы на Python

```
month = int(input("Введите номер месяца (от 1 до 12): "))
```

```
if month == 1 or month == 3 or month == 5 or month == 7 or month == 8 or month == 10 or month == 12:
```

```
    days = 31
```

```
elif month == 4 or month == 6 or month == 9 or month == 11:

    days = 30

elif month == 2:

    days = 28

else:

    days = "Неверный номер месяца."

print("Количество дней в месяце:", days)
```

| Протокол работы программы

1. Запуск программы:

- Пользователь запускает программу в среде PyCharm Community.

2. Ввод значения:

- Программа запрашивает ввод номера месяца.

3. Обработка входных данных:

- Программа проверяет, какой номер месяца ввел пользователь, и выполняет соответствующее ветвление с использованием конструкции `if...elif...else`.

4. Вывод результата:

- Программа отображает количество дней в месяце или сообщение о неверном вводе.

5. Завершение работы:

- Пользователь видит результат и завершается выполнение программы.

| Вывод

В целом, выполненные задачи способствовали углублению знаний в области программирования на Python, освоению принципов работы с ветвящимися структурами и развитию навыков алгоритмического мышления. Полученные знания и опыт будут полезны для дальнейшего изучения программирования и разработки более сложных приложений.