

Практическое занятие №3

Тема: составление программ ветвящейся структуры в IDE PyCharm Community.

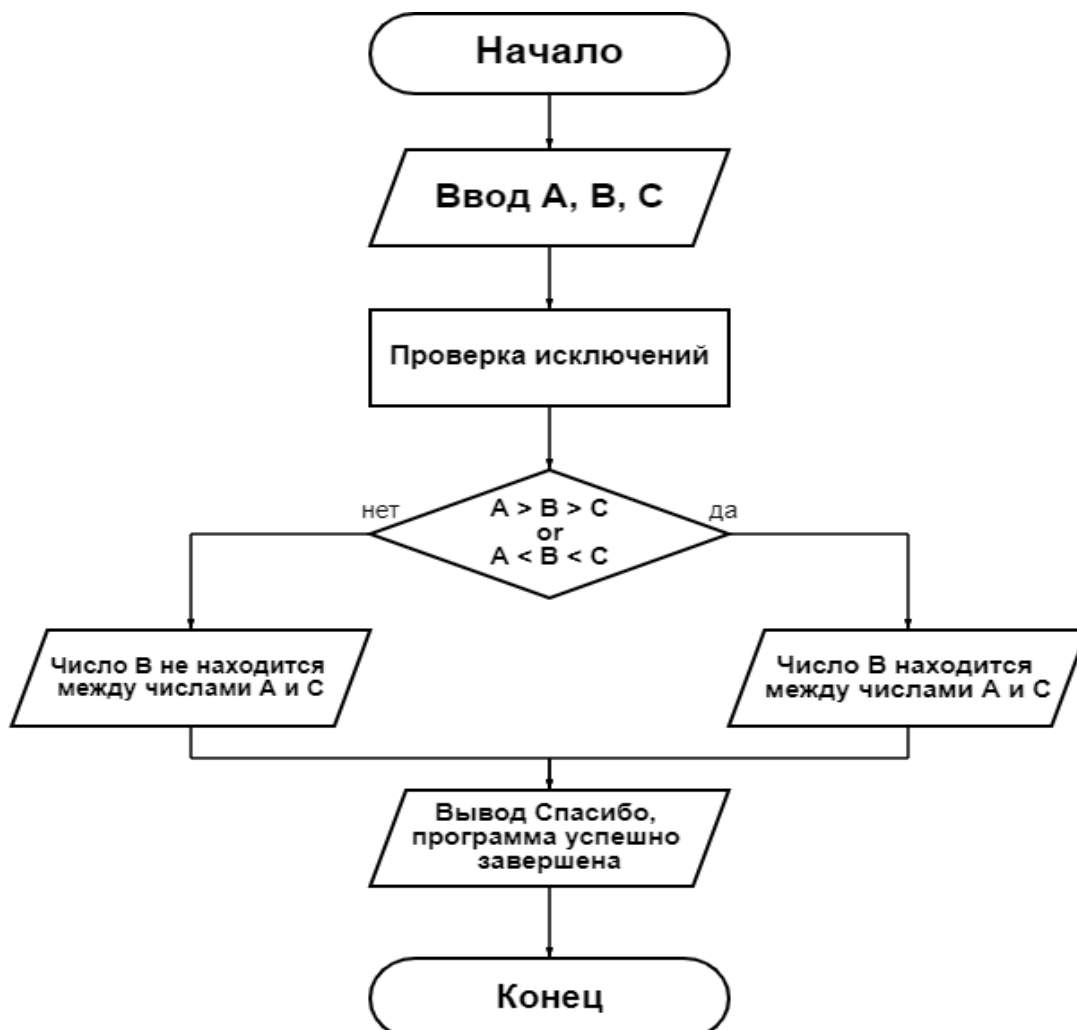
Цели практического занятия: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ ветвящейся структуры в IDE PyCharm Community.

Постановка задачи 1:

Даны три целых числа: A, B, C. Проверить истинность высказывания: «Число B находится между числами A и C».

Тип алгоритма: ветвление.

Блок-схема алгоритма №1:



Текст программы № 1:

```
# Даны три целых числа: А, В, С.
# Проверить истинность высказывания: "Число В находится
между числами А и С"
while True: # программа будет работать даже при ошибке.
    try: # пользователь ввёл числа.
        А, В, С = map(int, input("Введите три целых
числа:").split())
        if (А > В > С) or (А < В < С): # проверка числа В
на то, стоит ли оно между А и С.
            print("Число", В, "находится между числами",
А, "и", С)
            break # досрочно прерывает цикл
        else:
            print("Число", В, "не находится между
числами", А, "и", С)
            break # досрочно прерывает цикл
    except ValueError: # проверка исключений (пользователь
ввёл не число, на экран выводится текст, а не ошибка)
        print("Введите числа, пожалуйста!")
print("Спасибо, программа успешно завершена!")
```

Протокол работы программы №1:

```
Введите три целых числа:love love love
Введите числа, пожалуйста!
Введите три целых числа:-1 0 2
Число 0 находится между числами -1 и 2
Спасибо, программа успешно завершена!)

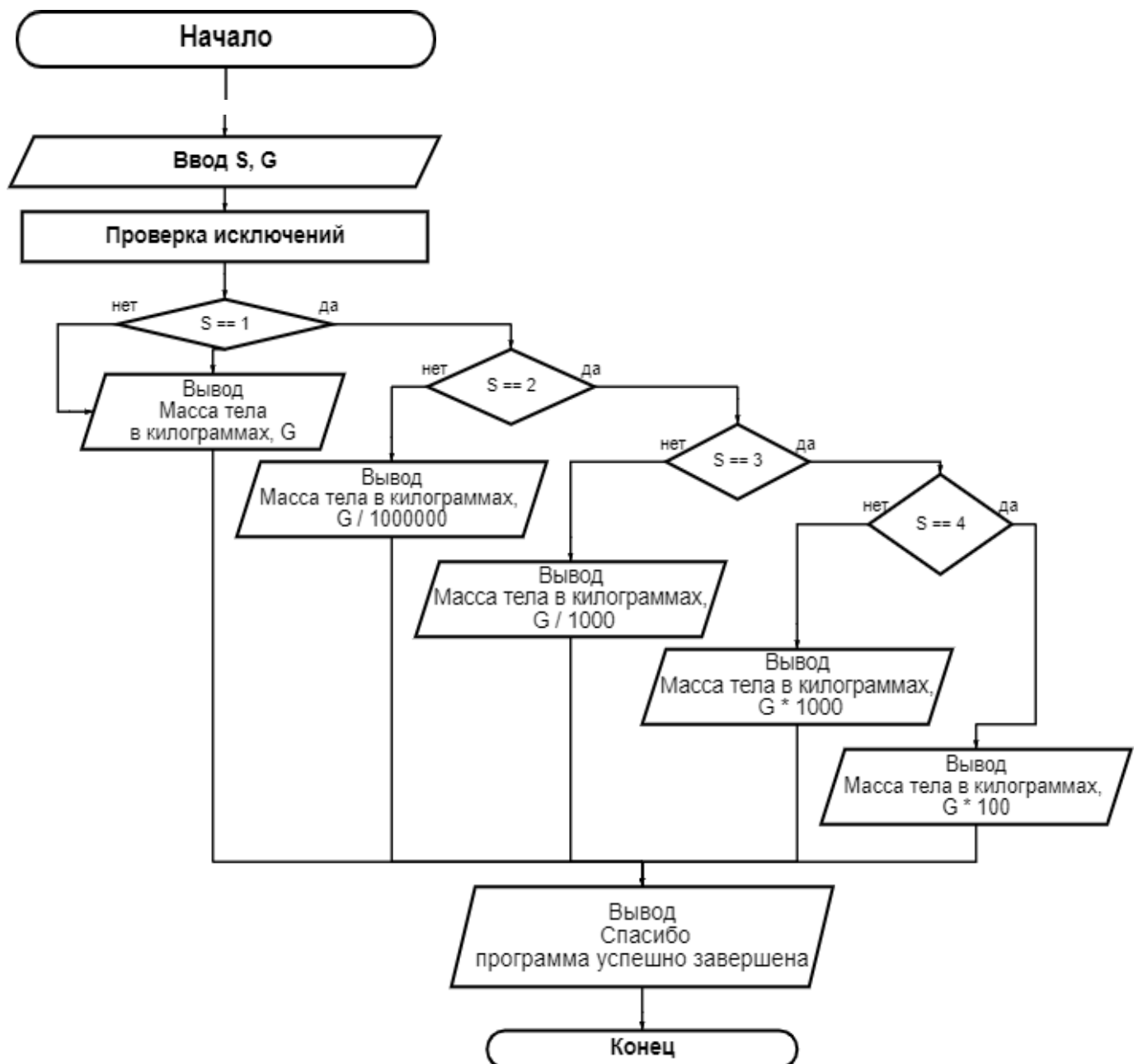
Process finished with exit code 0
|
```

Постановка задачи 2:

Единицы массы пронумерованы следующим образом: 1 – килограмм, 2 – миллиграмм, 3 – грамм, 4 – тонна, 5 – центнер. Дан номер единицы массы (целое число в диапазоне 1-5) и масса тела в этих единицах (вещественное число). Найти массу тела в килограммах.

Тип алгоритма: ветвление.

Блок-схема алгоритма №2:



Текст программы №2:

```
# Единицы массы пронумерованы следующим образом: 1 —  
килограмм, 2 — миллиграмм, 3 — грамм, 4 — тонна, 5 —  
центнер.  
# Дан номер единицы массы (целое число в диапазоне 1-5) и  
масса тела в этих единицах (вещественное число).  
# Найти массу тела в килограммах.  
while True: # программа постоянно работает даже при  
ошибке  
    try: # пользователь ввёл числа  
        print("Введите число от 1 до 5:")  
        print("1 - килограмм", "2 - миллиграмм", "3 -  
грамм", "4 - тонна", "5 - центнер", sep='\n')  
        S = int(input()) # для целочисленных значений  
        G = float(input("Введите массу тела в этих  
единицах:")) # для дробных значений  
        if S == 1: # выводим килограммы  
            print("Масса тела в килограммах = ", G)  
        elif S == 2: # переводим миллиграммы в килограммы  
            print("Масса тела в килограммах = ", G /  
1000000)  
        elif S == 3: # переводим граммы в килограммы  
            print("Масса тела в килограммах = ", G / 1000)  
        elif S == 4: # переводим тонны в килограммы  
            print("Масса тела в килограммах = ", G * 1000)  
        else: # переводим центнеры в килограммы  
            print("Масса тела в килограммах = ", G * 100)  
        print("Спасибо, программа успешно завершена!")  
        break # досрочно прерывает цикл  
    except ValueError: # проверка исключений (пользователь  
ввел не число, на экран выводится текст, а не ошибка)  
        print("Введите число, пожалуйста!")
```

Протокол работы программы №2:

```
Введите число от 1 до 5:  
1 - килограмм  
2 - миллиграмм  
3 - грамм  
4 - тонна  
5 - центнер  
5  
Введите массу тела в этих единицах:158  
Масса тела в килограммах = 15800.0  
Спасибо, программа успешно завершена!)
```

Process finished with exit code 0

Вывод:

Я закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрела навыки составления программ ветвящейся структуры в IDE PyCharm Community.

Были использованы языковые конструкции if, elif, else.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.