Практика

1. Виведіть на екран 10 зірочок завдяки циклу while.
2. Виведіть на екран 20 зірочок завдяки циклу do while
3. Виведіть на екран числа від 1 до 100 завдяки циклу while
4. Виведіть на екран числа від 100 до 0 завдяки циклу do while
5. Виведіть на екран ПАРНІ числа від 1 до 100 завдяки циклу while (і внутрішньому if).
6. Виведіть на екран НЕ ПАРНІ числа від 100 до 0 завдяки циклу do while.
7. Користувач вводить кількість чисел, які він буде вводити. Після цього він починає їх вводити цю кількість разів. Кожне число добавляється до суми. Скористайтесь циклом while і виведіть в кінці суму.
8. Користувач вводить числа до тих пір поки не введе 0. Кожного разу лічильник вводу збільшується на 1. Скористайтесь циклом do while і виведіть в кінці скільки чисел він ввів.
9. Користувач вводить 2 числа. Виведіть на екран всі числа між цими числами. Врахуйте помилку, якщо користувач спочатку введе більше а потім менше число. Використовуйте цикл while.
10. Користувач вводить ціле додатне число (більше за 0). Виведіть на екран це число в оберненому вигляді цифра за цифрою. Для цього скористайтесь циклом while. Підказка: вираз число % 10 дасть вам останню цифру. Вираз число /= 10 відкине від числа останню цифру. Ваше діло добавити вивід і придумати умову. Не забутьте добавити if, який перевірятиме, чи користувач дійсно ввів додатне число, яке більше за 0.
11. Та сама задача, тільки на екран виводиться кількість цифр в числі а не самі цифри. Поміняйте while на do while і зробіть так щоб враховувався 0 і від’ємні числа.
12. Користувач вводить число і ваша задача вивести це число по цифрам, тільки вже не в оберненому вигляді. Тобто якщо користувач введе 1889, то на екрані має бути 1889. Для цього вам треба буде порахувати спочатку скільки є цифр в числі, а потім число ділити на 10 в певній степені. Якщо не розумієте, чому, то спробуйте самі вирішити цю задачу. Щоб швидко піднести число до якогось степеню, напишіть на початку файлу код #include <cmath> - це підключить вам математичну бібліотеку, в якій є функція піднесення до степеню. Щоб нею скористатись, напишіть pow(число, степінь). Ця функція на місце свого виклику поверне готовий результат. Наприклад cout << pow(2, 3); виведе на екран 8. Які цикли вибирати – вирішуєте ви.
13. Користувач вводить число (обов’язково має бути додатне). Вивести на екран факторіал цього числа. Факторіал гугліть в неті. Це не складно. Факторіал числа N = 1\*2\*3\*…\*N.
14. У швачки є тканина довжиною L. Їй треба подушки довжиною Р. Домовимось, що ширина тканини така ж, як і ширина подушки. Визначити скільки подушок вона може пошити, якщо ділити і множити вона не вміє (використовувати лише цикл і оператори + або -)
15. На складі є деяка кількість яблук (яка вводить користувач). На склад приїжджають машини, які забирають по 400 яблук кожна. Вивести на екран необхідну кількість машин, щоб очистити склад.
16. Створіть перечислення Direction з константами HORIZONTAL, VERTICAL. Створіть змінну цього перечислення. Користувачу виводиться повідомлення «Виберіть лінію. H – горизонтальна, V – вертикальна». Користувач в символьну змінну вводить символ. Завдяки switch проаналізуйте його і якщо це один із правильних символів – задайте змінній перечислення відповідну константу. Інакше виведіть повідомлення про помилку і задайте значення яке хочете. Після цього спитайте в користувача скільки символів він хоче вивести. Користувач вводить цю кількість до тих пір поки не введе додатнє число більше за 0. Виконайте це в циклі do while. Після цього спитайте який символ він хоче вивести. Він його вводить. Після цього в switch проаналізуйте значення змінної перечислення, і якщо лінія має бути горизонтальною – виведіть рядок із певної кількості його символу. Якщо вертикальна – то стовпчик.
17. Ви вже створювати калькулятор. Але його проблемою було те, що після вибору операції він закінчував свою роботу. Верніться до тієї задачі (код у вас уже має бути або пишіть заново), і засуньте калькулятор в цикл do while. Тобто до пунктів меню добавите пункт виходу і пункт «Піднести до степеню». Використайте функцію, з якою познайомились в бібліотеці cmath. Разом з цим добавите пункт «Показати корінь». В цій же бібліотеці є функція sqrt яка приймає число і повертає його корінь, тобто вираз cout << sqrt(9); виведе на екран 3.