

Пояснительная записка

Название проекта: программа для вычисления даты католической Пасхалии.

Данные об авторе:

ВУЗ: НИУ Высшая Школа Экономики

Факультет: Факультет компьютерных наук

Направление: Программная инженерия

Группа: БПИ197

ФИО: Попов Олег Олегович

Условие:

Разработать программу вычисления даты католической Пасхалии для заданного года (вариант 17).

Методы и алгоритмы:

В основе программы лежит алгоритм Лилия-Клавия, состоящий из 11 этапов (далее Y обозначает год, в котором требуется определить дату Пасхи):

1. $G = (Y \bmod 19) + 1$ (G — так называемое «золотое число в Метоновом» цикле — 19-летнем цикле полнолуний)
2. $C = Y/100 + 1$ (если Y не кратен 100, то C — номер века)
3. $X = 3C/4 - 12$ (поправка на изъятие трёх из четырёх високосных вековых лет, «солнечное уравнение»)
4. $Z = (8C + 5)/25 - 5$ (поправка цикла Каллиппа, «лунное уравнение»)
5. $D = 5Y/4 - X - 10$ (в марте день — $D \bmod 7$ будет воскресенье)
6. $E = [(11G + 20 + Z - X) \bmod 30 + 30] \bmod 30$ (эпакта — указывает на день наступления полнолуния)
7. ЕСЛИ ($E = 24$) ИЛИ ($E = 25$ И $G > 11$), ТО увеличить E на 1
8. $N = 44 - E$ (N -е марта — день календарного полнолуния)
9. ЕСЛИ $N < 21$, ТО увеличить N на 30
10. $N = N + 7 - (D + N) \bmod 7$

11. ЕСЛИ $N > 31$, ТО дата Пасхи ($N - 31$) апреля, ИНАЧЕ дата Пасхи N марта

Элементы программы:

1. Макросы:

a. `devide`:

Определяет частное деления

Аргументы: `dividend` - делимое, `divider` - делитель, `quotient` - переменная, в которую будет записано частное.

b. `getRemainder`:

Определяет остаток от деления

Аргументы: `dividend` - делимое, `divider` - делитель, `quotient` - переменная, в которую будет записан остаток.

c. `multiply`:

Вычисляет произведение 2х целых чисел

Аргументы: `elem1` - 1-е число, `elem2` - 2-е число, `res` - переменная, в которую будет записано произведение.

2. Методы:

a. `YearInput`:

Осуществляет ввод года в переменную `year`, проверяет корректность ввода.

b. `CalculateDate`:

Производит вычисления даты католической Пасхи на основе алгоритма Лилия-Клавия. Номер дня месяца сохраняет в переменную `easterDay`, номер месяца - в `easterMonth`.

c. `PrintDate`:

Выводит дату католической Пасхи в консоль.

3. Основные переменные:

a. `year` - год (входные данные)

b. `easterDay` - номер дня Пасхи (выходные данные)

c. `easterMonth` - номер месяца Пасхи (выходные данные)

d. `G` - золотое число в Метоновом цикле

e. `C` - номер века

- f. X - поправка на изъятие 3-х из 4-х високосных вековых лет
- g. Z - поправка цикла Каллиппа
- h. D - в марте день под номером $D \bmod 7$ - воскресенье
- i. E - Эпакта - день наступления полнолуния
- j. N - день в марте, являющийся календарным полнолунием
- k. temp - переменная для промежуточных вычислений

Протокол взаимодействия:

На вход программе подается год n . э.. Программа запросит ввод данных после вывода строки "Year = ", далее будет ожидать ввод года через консоль. В случае некорректного ввода программа сообщит об этом выводом строки "Incorrect year = <год> (year must be > 0)".

В случае корректного ввода программа выполнит вычисления и выведет строку "Easter date is <дата>", где <дата> - дата католической Пасхи в формате "дд.мм.гггг".

Тестирование:

Year = -1 Incorrect year = -1 (year must be > 0)	Year = 2019 Easter date is 21.04.2019	Year = 2021 Easter date is 4.04.2021
Year = 2016 Easter date is 27.03.2016	Year = 2020 Easter date is 12.04.2020	Year = 2022 Easter date is 17.04.2022

Текст работы:

Текст программы (код на языке программирования fasm) расположен на веб ресурсе github.com. Полная ссылка:
<https://github.com/ShadyRoll/ComputingArchitecture/blob/master/MiniProject/HomeWork.asm>

Список используемых источников:

1. fasm - основное руководство (flat assembler 1.64 - мануал программера) // flatassembler.narod.ru URL: <http://flatassembler.narod.ru/fasm.htm> (дата обращения: 26.10.2020).
2. Календарь православной и католической Пасхи по годам // calendar.by URL: <http://calendar.by/content.php?id=19> (дата обращения: 24.10.2020)
3. Пасхалия // ru.wikipedia.org URL: <https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%81%D1%85%D0%B0%D0%BB%D0%B8%D1%8F> (дата обращения: 27.10.2020).
4. Программирование на языке ассемблера. Микропроект. Требования к оформлению. 2020-2021 уч.г. // softcraft.ru URL: <http://softcraft.ru/edu/comparch/tasks/mp01/> (дата обращения: 25.10.2020).
5. Разработка программ на ассемблере. Использование подпрограмм // softcraft.ru URL: <http://softcraft.ru/edu/comparch/practice/asm86/03-subprog/> (дата обращения: 25.10.2020).