ПЗ-05: **Указатели на функцию**

**Теория** – см. лекцию.

**Постановка задачи:**

В командной строке (СLI – Command line interface) реализовать программу-калькулятор, которая будет производить вычисление арифметических операций.

Числа и знак операции вводятся пользователем.

После выполнения вычисления программа не должна завершаться, а должна запрашивать новые данные для вычислений.

Завершение программы должно выполняться при вводе символа ‘0’ в качестве знака операции.

Если пользователь вводит неверный знак (не ‘0’, ‘+’, ‘-‘, ‘\*’, ‘/’), то программа должна сообщать ему об ошибке и снова запрашивать знак операции.

Также сообщать пользователю о невозможности деления на ноль, если он ввел 0 в качестве делителя.

**Решить задачу двумя способами:**

1. стандартно (с использованием switch-case)
2. с использованием указателя на функцию.

**Алгоритм решения:**

1. Запустить бесконечный цикл.
2. Запросить ввод знака операции.
3. Если был введен ‘0’, то прервать цикл операцией break.
4. Если был введен один из знаков арифметических операций, то в зависимости от того, какой знак был введен, выполнить соответствующее ему арифметическое действие. При этом при делении проверить не был ли введен ноль. Если это так, то вывести сообщение, иначе выполнить деление.
5. Иначе сообщить о неверно введенном знаке.

**Пример стандартного решения**

#include <stdio.h>

void main()

{

float x,y;

char sign='+';

while (sign != '0')

{

printf("Sign: "); //Знак операции

scanf("%c%\*c", &sign);

if (sign == '0') { break; }

if (sign == '+' || sign == '-'

|| sign == '\*' || sign == '/')

{

printf("x=");

scanf("%f%\*c", &x);

printf("y=");

scanf("%f%\*c", &y);

switch (sign)

{

case '+':

printf("%.2f\n", x+y);

break;

case '-':

printf("%.2f\n", x-y);

break;

case '\*':

printf("%.2f\n", x\*y);

break;

case '/':

if (y != 0) printf("%.2f\n", x/y);

else printf("Division by zero!\n"); // Деление на ноль!

}

}

else printf("The operation sign is specified incorrectly\n"); // Неверно указан знак операции

}

}