



**ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ И НАУКИ
ГОРОДА МОСКВЫ**
**Государственное бюджетное профессиональное
образовательное учреждение города Москвы**
«Колледж малого бизнеса № 4»
(ГБПОУ КМБ № 4)

Отчёт по практической работе №1

Специальность: 09.02.07 Информационные системы и программирование

Форма обучения: очная

Студент: Межибор Ярослав Евгеньевич

Группа: ИПО-21.24

Проверил: Рыбаков Александр Сергеевич

Москва, 2025 г.

Вариант 5

Математическая модель задачи:

Приложение предназначено для решения двух независимых вычислительных задач, реализованных на разных вкладках.

А. Линейный алгоритм

Элемент	Описание	Формула	Ограничения
Выражение	Вычисление значения а	$a = \ln(y^2)$	x
Вход	Три действительных числа: x, y, z	-	-
Выход	Одно действительное число: а	-	-

Б. Разветвляющийся алгоритм

Элемент	Описание	Формула	Ограничения
Выражение	Вычисление значения k в зависимости от произведения xy и выбранной функции f(x).	$k = \begin{cases} (f(x) + y)^2, & 4 > xy > 1 \\ f(x) * \operatorname{tg}(y), & 8 < xy < 10 \\ f(x) + y, & \text{иначе} \end{cases}$	$y \neq \frac{\pi}{2} + \pi n, n \in \mathbb{Z}$ (для $\operatorname{tg}(y)$)
Функции f(x)	Выбор одной из трех функций	$f(x) \in \{\cos(x), x^2, e^x\}$	-
Вход	Два действительных числа: x, y. Выбор f(x).	-	-
Выход	Одно действительное число: k или сообщение об ошибке/области "иначе".	-	-

Спецификация программы:

Программа является Windows-приложением, состоящим из двух функциональных вкладок, обеспечивающих пользовательский ввод и вывод результатов.

А. Спецификация вкладки Линейный алгоритм

- Входные данные:** Предоставить пользователю три текстовых поля для ввода числовых значений X, Y и Z.

- **Функциональность:**
 1. Принять значения X, Y, Z.
 2. Проверить **область определения**: если $Y = < 0$, вывести сообщение об ошибке.
 3. Выполнить вычисление α по заданной формуле.
- **Выходные данные:** Вывести результат α в отдельное поле результата.

Б. Спецификация вкладки Разветвляющийся алгоритм

- **Входные данные:**
 1. Два текстовых поля для ввода чисел X и Y.
 2. Набор переключателей (Radio Buttons) для выбора функции $f(x)$ ($\cos(x)$, x^2 , e^x).
- **Функциональность:**
 1. Вычислить произведение $P = X * Y$.
 2. Применить соответствующую формулу в зависимости от того, в какой диапазон попало P.
 3. Если P попадает в область "иначе" (т.е. $P < 1$, $P \geq 4$, $P = < 8$ или $P \geq 10$), вывести соответствующее сообщение.
- **Выходные данные:** Вывести результат k или сообщение о попадании в область "иначе".

Алгоритм программы:

А. Алгоритм для линейного алгоритма (Вычисление α)

1. Начало.
2. Ввод X, Y, Z.
3. Если $Y = < 0$, то вывести "Ошибка: Y должно быть > 0 " и Конец.
4. Вычисление: $\alpha = \ln(y^{-\sqrt{|x|}}) \cdot (x - \frac{y}{2}) + (\sin(\arctg(z)))^2$
5. Вывод результата α
6. Конец.

Б. Алгоритм для разветвляющегося алгоритма (Вычисление k)

1. Начало.
2. Ввод X, Y. Определение выбранной функции $f(x)$.
3. Вычисление произведения $P = X * Y$.
4. Если $1 = < P$ И $P < 4$:
 - $k = (f(x) + Y)^2$.
5. Иначе, если $8 < P$ И $P < 10$:
 - Проверка $\operatorname{tg}(y)$: Если Y близко к $\pi/2 + \pi n$ вывести "Ошибка: Недопустимое значение Y" и Конец.
 - $k = f(x) \cdot \operatorname{tg}(Y)$
6. Иначе (любое другое значение P):
 - $k = \text{"Значение XY вне заданных диапазонов [1, 4) и (8, 10]".}$
7. Вывод результата k.
8. Конец.

Отладка логики программы:

Процесс:

1. Использовался метод ручного тестирования при котором разработчик (то есть я) пошагово выполняет алгоритм, а сосед (проверяющий) независимо рассчитывает ожидаемый результат.
2. Сравниваются промежуточные и конечные значения.

Ключевой результат отладки:

- Обнаруженная ошибка: В исходном алгоритме линейной задачи отсутствовала явная проверка области определения логарифма. При ручном прогоне на тестовом наборе с $Y=-1$ была обнаружена логическая ошибка (попытка вычислить $\ln(-1)$ или $\ln(y^{-\sqrt{|x|}})$, где $y \leq 0$)
- Исправление: В алгоритм был добавлен обязательный шаг: Проверка $Y > 0$. Это гарантирует устойчивость приложения и предотвращает ошибки выполнения.

Тестовые наборы для проверки функционала системы:

A. Проверка Линейного алгоритма

№	X	Y	Z	Ожидаемый результат a	Цель теста
1	4	e	0	~ -5.282	Основной расчёт
2	0	1	1	0.5	Проверка x=0, $\ln(y^0)$
3	1	-1	1	Ошибка	Проверка ограничения $Y > 0$
3	1	-1	1	Ошибка	Проверка ограничения $Y > 0$

Б. Проверка Разветвляющегося алгоритма (Полное покрытие условий и функций)

№	X	Y	P=X·Y	f(x)	Ожидаемый результат k	Проверяемое условие
4	2	1.5	3	x^2	30.25	$1 = < P < 4$
5	1	1	1	$\cos(x)$	$(\cos(1)+1)^2$ ~2.459	$P=1$ (Нижняя граница)
6	1.9	2	3.8	e^x	$(e^{1.9}+2)^2$ ~76.56	$P \rightarrow 4$ (Верхняя граница)
7	10	0.9	9	$\cos(x)$	~-1.057	$8 < P < 10$

8	8.1	1	8.1	x^2	$8.1^{12} * \operatorname{tg}(1)$ ~ 102.16	$P > 8$ (Нижняя граница)
9	5	2	10	e^x	"Вне диапазона"	$P=10$ (Граница "иначе")
10	1	5	5	$\cos(x)$	"Вне диапазона"	Область "иначе"
11	1	$\pi/2$	$\pi/2 \sim 1.57$	x^2	Ошибка	Проверка ограничения $\operatorname{tg}\{y\}$ (для $P > 8$)

Контрольные вопросы:

1. Ручная отладка (метод ручного тестирования) – это самый простой способ отладки, заключающийся в ручном пошаговом выполнении (прогоне) программы, используя тот же набор входных данных, при котором была обнаружена ошибка. Это позволяет программисту проследить за логикой исполнения и локализовать конкретный оператор, вызвавший нарушение.
2. Ручная отладка проводится на этапе отладки, после того, как тестирование выявило ошибку. В общей методике отладки это соответствует 2-му (определение локализации ошибки) и 3-му (определение причины ошибки) этапам.
3. Отладка всегда требует логического осмыслиения информации об ошибке. Основные методы, используемые для обнаружения ошибок по косвенным признакам, включают:
 - Метод ручного тестирования (ручной отладки) : Пошаговый ручной прогон программы для поиска ошибки.
 - Индукция: Выдвижение общей гипотезы о причине ошибки, основанное на наблюдениях за конкретными проявлениями.
 - Дедукция: Формирование списка возможных причин и их систематическая проверка и исключение.
 - Обратное прослеживание: Анализ выполнения программы в обратном порядке от точки проявления ошибки к исходному коду.

Вывод:

В ходе выполнения практической работы была изучена теория отладки программного обеспечения и освоен метод ручной отладки (метод "грубой силы"). Были описаны математическая модель и спецификация программы, а также записаны алгоритмы для линейного и разветвляющегося алгоритмов, заданных по варианту №5. Составлены тестовые наборы, обеспечивающие проверку всех ветвей разветвляющегося алгоритма и пограничных/ограничивающих условий линейного алгоритма. В процессе ручного прогона была обнаружена и локализована логическая ошибка (отсутствие проверки $Y > 0$ в линейном алгоритме), что подтвердило эффективность ручного метода в выявлении логических ошибок до запуска программы.