Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
(Университет ИТМО)

Факультет безопасности информационных технологий

Дисциплина:

«Статистические методы в инженерных исследованиях»

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

«Первичная обработка экспериментальных данных.
Проверка гипотез о значениях параметров распределения: критерии значимости»

Вариант 5

Студент группы N3346 ______________ Бардышев А. А.
(подпись)

Студент группы N3346 ______________ Волощук А. Н.
(подпись)

Студент группы N3346 ______________ Замахов Е. В.
(подпись)

Студент группы N3346 ______________ Пинус И. В.
(подпись)

Студент группы N3346 ______________ Суханкулиев М.
(подпись)

Студент группы N3346 ______________ Шегай С. Д.
(подпись)

Проверил:

Преподаватель, д.т.н., профессор ______________ Федоров А. В.
(подпись)

«___» ________________ 2025 г.

Санкт-Петербург

2025 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Цель работы – получить умения и отработать навыки проверки гипотез о значениях параметров распределения.

Для достижения поставленной цели необходимо решить следующие задачи:

- провести серию экспериментов (измерений);
- обработать результаты измерений и провести проверки гипотез о значениях параметров распределения;
- оформить отчет по лабораторной работе.

1 АНАЛИЗ ВЫБОРКИ ТВЕРДОСТИ МАТЕРИАЛА (ЗАДАЧА 1)

1.1 Постановка задачи и исходные данные

В данном разделе требуется по результатам 50 независимых равноточных измерений твердости материала (НВ) определить:

1. Истинную твердость металлической пластины, предварительно исключив грубые ошибки (выбросы) при доверительной вероятности $p = 0.95$.
2. Доверительные интервалы для математического ожидания и среднеквадратического отклонения (СКО) при доверительной вероятности $p = 0.95$.
3. Минимальное число измерений, необходимое для того, чтобы с вероятностью $p = 0.95$ предельная погрешность оценки истинной твердости не превышала $\varepsilon = 10\ HB$.

Исходные данные для анализа представлены в таблице 1.

Таблица 1 – Исходные экспериментальные данные, НВ

Вариант 5				
71	79	89	66	76
69	66	96	50	58
89	66	68	75	64
70	66	54	69	71
71	72	70	52	55
51	82	68	87	69
80	70	42	72	81
93	62	46	76	85
73	64	63	63	75
61	62	62	57	61

1.2 Исключение грубых ошибок измерений

Перед расчетом основных характеристик необходимо очистить выборку от грубых ошибок (выбросов). Для этого воспользуемся методом, основанным на t-распределении Стьюдента, при доверительной вероятности $p = 0.95$ (уровень значимости $\alpha = 0.05$). Процедура является итерационной: на каждом шаге вычисляется наблюдаемое значение критерия $t_{\text{набл}}$ для наиболее подозрительного значения и сравнивается с критическим $t_{\text{кр}}$.

```
Итерация 1:
n = 50, M(x) = 68.74, S(x) = 11.77.
Подозрительное значение: x_max = 96.
t_набл = |68.74 - 96| / 11.77 = 2.32.
t_кр(0.95; ν=48) = 1.68.
Вывод: Так как 2.32 > 1.68, значение 96 HB является выбросом и удаляется.
Итерация 2:
n = 49, M(x) = 68.18, S(x) = 11.21.
Подозрительное значение: x_min = 42.
t_набл = |68.18 - 42| / 11.21 = 2.34.
t_кр(0.95; ν=47) = 1.68.
Вывод: Так как 2.34 > 1.68, значение 42 HB является выбросом и удаляется.
```

...последующие итерации аналогично выявляли и удаляли выбросы...

```
Итерация 24:
n = 27, M(x) = 67.00, S(x) = 3.80.
Подозрительное значение: x_min = 61.
t_набл = |67.00 - 61| / 3.80 = 1.58.
t_кр(0.95; ν=25) = 1.71.
Вывод: Так как 1.58 <= 1.71, нулевая гипотеза об отсутствии выбросов
      принимается. Процедура очистки завершена.
```

В результате итерационной процедуры из исходной выборки было исключено 23 значения, признанных грубыми ошибками. Дальнейшие расчеты будут производиться на основе очищенной выборки объемом $n = 27$.

Следует отметить, что в ходе итерационной процедуры было удалено 23 из 50 наблюдений (46%), что является необычно высоким показателем. Это может свидетельствовать о том, что исходное распределение твердости существенно отличается от нормального, из-за чего примененный метод, рассчитанный на симметричные данные, оказался излишне «агрессивным». Тем не менее, в рамках данной работы процедура была выполнена в соответствии с заданной методикой.

1.3 Определение доверительных интервалов

Расчеты производятся для очищенной выборки объемом $n = 27$, для которой точечные оценки параметров составляют:

– Среднее значение (оценка мат. ожидания): $M(x) = 67.00$ HB.

– Стандартное отклонение (оценка СКО): $S(x) = 3.80$ HB.

1. Доверительный интервал для математического ожидания (истинной твердости)

Интервал рассчитывается по формуле:

$$M(x) - t_{\text{кр}} \cdot \left(\frac{S(x)}{\sqrt{n}}\right) < \mu < M(x) + t_{\text{кр}} \cdot \left(\frac{S(x)}{\sqrt{n}}\right)$$

где $t_{кр}$ – критическое значение критерия Стьюдента для двусторонней критической области при уровне значимости $\alpha = 1 - p = 0.05$ и числе степеней свободы $\nu = n - 1 = 26$.

$$t_{кр}(\alpha = 0.05; \nu = 26) = 2.056$$

Предельная погрешность оценки: $\varepsilon = 2.056 \cdot \left(\frac{3.80}{\sqrt{27}}\right) \approx 1.50\ HB$.

– Нижняя граница: $67 - 1.5 = 65.5\ HB$.

– Верхняя граница: $67 + 1.5 = 68.5\ HB$.

Таким образом, с доверительной вероятностью 95% истинное значение твердости материала находится в интервале от 65.50 HB до 68.50 HB.

2. Доверительный интервал для среднеквадратического отклонения

Интервал для СКО рассчитывается на основе распределения хи-квадрат (χ^2) по формуле:

$$S(x) \cdot \sqrt{\frac{n-1}{\chi^2_{пр}}} < \sigma < S(x) \cdot \sqrt{\frac{n-1}{\chi^2_{лев}}}$$

где $\chi^2_{пр}$ и $\chi^2_{лев}$ – правая и левая критические точки распределения хи-квадрат для $\alpha = 0.05$ и $\nu = n - 1 = 26$.

$$\chi^2_{пр} = \chi^2\left(\frac{\alpha}{2}; \nu\right) = \chi^2(0.025; 26) = 41.92$$

$$\chi^2_{лев} = \chi^2\left(\frac{1-\alpha}{2}; \nu\right) = \chi^2(0.975; 26) = 13.84$$

– Нижняя граница: $3.8 \cdot \sqrt{\frac{26}{41.92}} \approx 2.99\ HB$.

– Верхняя граница: $3.8 \cdot \sqrt{\frac{26}{13.84}} \approx 5.21\ HB$.

Таким образом, с доверительной вероятностью 95% истинное значение СКО находится в интервале от 2.99 HB до 5.21 HB.

1.4 Расчет необходимого объема выборки

В данном пункте необходимо определить минимальное число измерений n_{min}, которое обеспечит предельную погрешность ε точечной оценки истинной твердости не более 10 HB при доверительной вероятности $p = 0.99$ (уровень значимости $\alpha = 0.01$).

Для расчета используется неравенство:

$$n > \left(S(x) \cdot t_{кр}(\alpha; n-1)/\varepsilon\right)^2$$

где в качестве оценки стандартного отклонения $S(x)$ мы используем значение, полученное по нашей очищенной выборке: $S(x) = 3.8\,HB$.

Поскольку $t_{кр}$ зависит от искомого n, задача решается итерационно.

Итерация 1: Примем $n_0 = 27$ (текущий размер).

$$t_{кр}(0.01; 26) = 2.779$$

Правая часть неравенства: $(3.80 \cdot 2.779 \ / \ 10)^2 \ \approx \ 1.11$.

Условие $27 \ > \ 1.11$ выполняется с огромным запасом. Это означает, что для достижения такой низкой точности (погрешность целых 10 HB) достаточно гораздо меньшего числа измерений.

Поиск минимального n: Проведем итерационный подбор, начиная с $n = 2$.

При $n = 2$: $t_{кр}(0.01; \ 1) = 63.657$.

Правая часть: $(3.80 \cdot \ 63.657 \ / \ 10)^2 \ \approx \ 584.4.2 \ < \ 586.$

…

При $n = 5$: $t_{кр}(0.01; \ 4) \ = \ 4.604$.

Правая часть: $(3.80 \cdot \ 4.604 \ / \ 10)^2 \ \approx \ 3.06.5 \ > \ 3.06.$

Условие $n \ > $ правая_часть впервые выполняется при $n = 5$.

Вывод: Для того чтобы с доверительной вероятностью 99% утверждать, что предельная погрешность точечной оценки истинной твердости не превышает 10 HB, необходимо выполнить минимум 5 измерений.

2 ПРОВЕРКА НАСТРОЙКИ УСТАНОВКИ (ЗАДАЧА 2)

2.1 Постановка задачи и исходные данные

В данном разделе требуется проверить правильность настройки установки для раскатки металла на основе $n = 20$ независимых равноточных измерений толщины листового проката.

Для варианта 5 установлены следующие технологические параметры:

– Номинальная толщина (целевое значение): $h_{\text{ном}} = M_0 = 50.0$ мм.

– Симметричный допуск: $\Delta h = \pm 1.0$ мм.

Необходимо по результатам измерений определить, соответствует ли средняя фактическая толщина проката номинальному значению. Исходные данные для анализа представлены в таблице 2.

Таблица 2 – Результаты измерений толщины проката, мм

№	**h**, мм	**Δh**, мм	**$h_{\text{узк}}$**, мм																			
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
5	**50,0**	**1,0**	50,7	50,8	50,6	51,1	49,4	49,8	50,7	49,9	50,1	50,6	50,8	52,0	50,5	49,7	49,7	51,6	49,7	50,5	50,4	50,1

2.2 Проверка гипотезы о значении математического ожидания

Для проверки правильности настройки установки необходимо проверить нулевую гипотезу H_0 о равенстве истинного среднего значения (математического ожидания μ) толщины проката номинальному значению $M_0 = 50.0$ мм.

– Нулевая гипотеза H_0: $\mu = 50.0$ (установка настроена верно).

– Альтернативная гипотеза H_1: $\mu \neq 50.0$ (установка настроена неверно).

Поскольку дисперсия генеральной совокупности неизвестна, для проверки гипотезы применяется t-критерий Стьюдента. Уровень значимости примем $\alpha = 0.05$.

Сначала рассчитаем выборочные характеристики:

– Объем выборки: $n = 20$.

– Выборочное среднее: $M(h) = 50.44$ мм.

– Выборочное стандартное отклонение: $S(h) = 0.66$ мм.

Наблюдаемое значение t-статистики рассчитывается по формуле:

$$t_{\text{набл}} = \frac{|M(h) - M_0|}{S(h)/\sqrt{n}} = \frac{|50.44 - 50.0|}{0.66/\sqrt{20}} \approx 2.94$$

Критическое значение $t_{\text{кр}}$ для двусторонней критической области находится по таблицам Стьюдента при уровне значимости $\alpha = 0.05$ и числе степеней свободы $\nu = n - 1 = 19$.

$$t_{\text{кр}}(\alpha = 0.05;\ \nu = 19) = 2.093$$

Сравнение и вывод:

Так как $t_{\text{набл}} = 2.94 > t_{\text{кр}} = 2.093$, нулевая гипотеза H_0 отвергается. Этот вывод также подтверждается p-значением ($p - value$), вычисленным программно в ходе выполнения t-теста. Полученное $p - value = 0.0084$ значительно меньше уровня значимости $\alpha = 0.05$, что указывает на высокую статистическую значимость расхождения.

Заключение по задаче: Полученное расхождение между выборочным средним (50.44 мм) и номинальным значением (50.0 мм) является статистически значимым. Это означает, что с вероятностью 95% можно утверждать, что установка настроена неправильно и производит прокат, систематически превышающий заданную толщину. Установка нуждается в регулировке.

3 СРАВНИТЕЛЬНАЯ ОЦЕНКА ЭФФЕКТИВНОСТИ ПО (ЗАДАЧА 3)

3.1 Постановка задачи и исходные данные

В данном разделе проводится статистический анализ результативности модернизации программного обеспечения (ПО) для парирования угроз. Цель – определить, приводит ли модернизация к статистически значимому увеличению количества парированных угроз.

По условию варианта №5, сравнению подлежат три исполнения ПО:

- Базовое (Б)
- Исполнение 2 (Исп. 2)
- Исполнение 4 (Исп. 4)

Данные были получены в трех независимых лабораториях, в каждой из которых проводилось по 12 испытаний для каждого типа ПО (таблица 3).

Таблица 3 – Исходные данные по лабораториям (количество парированных угроз)

Лаборатория 1	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Исп. Б	8	11	10	9	8	8	10	13	9	7	9	9
Исп. 2	11	15	11	10	10	12	11	11	11	12	14	12
Исп. 4	12	11	13	9	14	12	9	12	13	15	15	14
Лаборатория 2	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Исп. Б	11	11	12	11	7	8	8	11	9	9	11	9
Исп. 2	13	9	9	12	10	9	11	10	12	11	12	11
Исп. 4	14	13	13	10	13	13	12	9	15	13	11	15
Лаборатория 3	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Исп. Б	10	8	12	10	10	12	9	11	12	8	13	9
Исп. 2	10	11	10	12	13	9	9	13	11	11	10	10
Исп. 4	13	15	9	11	10	12	11	14	8	11	12	11

Для получения обобщенной оценки эффективности данные из трех лабораторий были объединены в три соответствующие выборки. Таким образом, объем каждой выборки составляет $n = 12 \cdot 3 = 36$.

3.2 Сравнительный анализ эффективности

Для ответа на вопрос о результативности модернизации проведем попарное сравнение средних значений выборок с помощью t-критерия Стьюдента для независимых выборок. В более строгом исследовании для сравнения трех и более групп следовало бы применить дисперсионный анализ (ANOVA) с последующими post-hoc тестами для коррекции на множественные сравнения, однако в рамках данной работы мы ограничимся прямыми попарными тестами. Уровень значимости для всех тестов примем $\alpha = 0.05$.

3.2.1 Сравнение: Базовое ПО vs Исполнение 2

1. Предварительные статистики:

– Базовое (Б): $n = 36, M(x) = 9.78, S(x) = 1.64$.

– Исп. 2: $n = 36, M(x) = 11.06, S(x) = 1.43$.

2. Проверка гипотезы о равенстве дисперсий (F-критерий Фишера):

H_0: $\sigma^2_{\text{Б}} = \sigma^2_{\text{Исп2}}$.

$F_{\text{набл}} = S^2_{\text{Б}}/S^2_{\text{Исп2}} = (1.64)^2 / (1.43)^2 \approx 1.31$.

Критическое значение $F_{\text{кр}}(\alpha = 0.05;\ 35;\ 35) \approx 1.76$.

Вывод: Так как $F_{\text{набл}} = 1.31 < F_{\text{кр}} = 1.76$, гипотеза о равенстве дисперсий не отвергается.

3. Проверка гипотезы о равенстве средних (t-критерий для равных дисперсий):

H_0: $\mu_{\text{Б}} = \mu_{\text{Исп2}}$ (эффективность не изменилась).

H_1: $\mu_{\text{Б}} < \mu_{\text{Исп2}}$ (эффективность Исп. 2 значимо выше).

$t_{\text{набл}} \approx -3.52$.

Критическое значение для одностороннего теста $t_{\text{кр}}(\alpha = 0.05;\ \nu = 70) \approx -1.67$.

Вывод: Так как $t_{\text{набл}} = -3.52 < t_{\text{кр}} = -1.67$, нулевая гипотеза H_0 отвергается.

Заключение: Модернизация до Исполнения 2 привела к статистически значимому увеличению количества парированных угроз.

3.2.2 Сравнение: Базовое ПО vs Исполнение 4

1. Предварительные статистики:

– Базовое (Б): $n = 36, M(x) = 9.78, S(x) = 1.64$.

– Исп. 4: $n = 36, M(x) = 12.14, S(x) = 1.96$.

2. Проверка гипотезы о равенстве дисперсий (F-критерий Фишера):

$H_0: \sigma^2_{Б} = \sigma^2_{Исп4}$.

$F_{набл} = S^2_{Исп4}/S^2_{Б} = (1.96)^2 / (1.64)^2 \approx 1.43$.

Критическое значение $F_{кр}(\alpha = 0.05;\ 35;\ 35) \approx 1.76$.

Вывод: Так как $F_{набл} = 1.43 < F_{кр} = 1.76$, гипотеза о равенстве дисперсий не отвергается.

3. Проверка гипотезы о равенстве средних (t-критерий для равных дисперсий):

$H_0: \mu_{Б} = \mu_{Исп4}$.

$H_1: \mu_{Б} < \mu_{Исп4}$.

$t_{набл} \approx -5.54$.

Критическое значение $t_{кр}(\alpha = 0.05;\ \nu = 70) \approx -1.67$.

Вывод: Так как $t_{набл} = -5.54 < t_{кр} = -1.67$, нулевая гипотеза H_0 отвергается.

Заключение: Модернизация до Исполнения 4 привела к статистически значимому увеличению количества парированных угроз.

3.2.3 Сравнение: Исполнение 2 vs Исполнение 4

1. Предварительные статистики:

– Исп. 2: $n = 36, M(x) = 11.06, S(x) = 1.43$.
– Исп. 4: $n = 36, M(x) = 12.14, S(x) = 1.96$.

2. Проверка гипотезы о равенстве дисперсий (F-критерий Фишера):

$H_0: \sigma^2_{Исп2} = \sigma^2_{Исп4}$.

$F_{набл} = S^2_{Исп4}/S^2_{Исп2} = (1.96)^2 / (1.43)^2 \approx 1.87$.

Критическое значение $F_{кр}(\alpha = 0.05;\ 35;\ 35) \approx 1.76$.

Вывод: Так как $F_{набл} = 1.87 > F_{кр} = 1.76$, гипотеза о равенстве дисперсий отвергается. Дисперсии считаются неоднородными.

3. Проверка гипотезы о равенстве средних (t-критерий для неравных дисперсий – критерий Уэлча):

$H_0: \mu_{Исп2} = \mu_{Исп4}$.

$H_1: \mu_{Исп2} \neq \mu_{Исп4}$ (двусторонняя гипотеза).

$t_{набл} \approx -2.68$.

Критическое значение для двустороннего теста $t_{кр}(\alpha = 0.05;\ \nu \approx 65) \approx \pm 2.00$.

Вывод: Так как $|t_{набл}| = 2.68 > t_{кр} = 2.00$, нулевая гипотеза H_0 отвергается.

Заключение: Существует статистически значимая разница в эффективности между Исполнением 2 и Исполнением 4.

3.3 Итоговые выводы по задаче 3

На основании проведенного статистического анализа можно сделать следующие выводы:

– Обе модернизации (до Исполнения 2 и до Исполнения 4) являются результативными, так как приводят к статистически значимому увеличению среднего количества парированных угроз по сравнению с базовой версией ПО.

– Исполнение 4 является статистически значимо более эффективным, чем Исполнение 2.

Таким образом, с точки зрения повышения эффективности, наиболее оправданной является модернизация до Исполнения 4.

ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы была достигнута поставленная цель: получены практические умения и отработаны навыки проверки статистических гипотез о значениях параметров распределения с использованием современных программных средств.

В рамках решения поставленных задач были получены следующие ключевые результаты:

1. Проведен анализ выборки твердости материала: на основе итерационной процедуры были выявлены и исключены 23 выброса. Для очищенной выборки определены доверительные интервалы: с вероятностью 95% истинное значение твердости находится в диапазоне от 65.50 НВ до 68.50 НВ.
2. Выполнена проверка настройки технологического оборудования. Статистический анализ показал, что средняя толщина производимого проката (50.44 мм) статистически значимо отличается от номинального значения (50.0 мм). Сделан вывод о необходимости регулировки установки.
3. Проведена сравнительная оценка эффективности трех версий программного обеспечения. Установлено, что обе модернизации (Исполнение 2 и Исполнение 4) являются результативными. При этом Исполнение 4 показало статистически значимо более высокую эффективность, чем Исполнение 2, и является наиболее предпочтительным для внедрения.

Все задачи лабораторной работы успешно решены, что позволило закрепить теоретические знания о критериях значимости и применить их для решения практических инженерных задач.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Сидняев, Н. И. Теория планирования эксперимента и анализ статистических данных : учебное пособие для вузов / Н. И. Сидняев. – 2-е изд., перераб. и доп. – Москва : Издательство Юрайт, 2011. – 399 с. – Текст : непосредственный.
2. Рыков, В. В. Математическая статистика и планирование эксперимента : учебное пособие / В. В. Рыков, В. Ю. Иткин. – Москва : РГУ нефти и газа им. И. М. Губкина, 2008. – 210 с. – Текст : непосредственный.
3. ГОСТ 24026-80. Исследовательские испытания. Планирование эксперимента. Термины и определения. – Введ. 1981-07-01. – Москва : Издательство стандартов, 1980. – 15 с. – Текст : непосредственный.
4. NumPy: the fundamental package for scientific computing with Python : официальный сайт. – Текст : электронный. – 2025. – URL: https://numpy.org/doc/stable/ (дата обращения: 15.10.2025).
5. Pandas: powerful Python data analysis toolkit : официальный сайт. – Текст : электронный. – 2025. – URL: https://pandas.pydata.org/docs/ (дата обращения: 15.10.2025).
6. Varoquaux, G. SciPy: high-level scientific computing / G. Varoquaux, A. Chauve, A. Espaze [et al.]. – Текст : электронный // Scientific Python Lecture Notes. – 2025. – URL: https://main--lectures-scientific-python-org.netlify.app/intro/scipy/ (дата обращения: 15.10.2025).
7. Matplotlib: Visualization with Python : официальный сайт. – Текст : электронный. – 2025. – URL: https://matplotlib.org/stable/contents.html (дата обращения: 15.10.2025).
8. Statsmodels: Statistic in Python : официальный сайт. – Текст : электронный. – 2025. – URL: https://www.statsmodels.org/stable/index.html (дата обращения: 16.10.2025).

ПРИЛОЖЕНИЕ А

(обязательное)

Листинг А.1 – Код для выполнения расчетов в среде Google Colaboratory

```
# --- Импорт библиотек ---
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy import stats

# --- Задача 1. Анализ выборки твердости материала ---
print("--- Задача 1. Анализ выборки твердости материала ---")

# 1.1. Исходные данные
data_task1 = np.array([
    71, 79, 89, 66, 76, 69, 66, 96, 50, 58, 89, 66, 68, 75, 64,
    70, 66, 54, 69, 71, 71, 72, 70, 52, 55, 51, 82, 68, 87, 69,
    80, 70, 42, 72, 81, 93, 62, 46, 76, 85, 73, 64, 63, 63, 75,
    61, 62, 62, 57, 61
])

print(f"Исходный объем выборки n = {len(data_task1)}")

# 1.2. Исключение грубых ошибок

# Копируем исходные данные, чтобы не портить оригинал
data_clean = data_task1.copy()
p_outlier = 0.95
alpha_outlier = 1 - p_outlier

print(f"--- 1.2. Исключение грубых ошибок (p={p_outlier}) ---")

iteration = 1
while True:
    n = len(data_clean)
    if n < 3: # Нельзя применять критерий для слишком маленьких выборок
        print("Объем выборки слишком мал для продолжения.")
        break

    mean_val = np.mean(data_clean)
    std_val = np.std(data_clean, ddof=1)

    # Находим наиболее удаленное значение
    x_min_val = data_clean.min()
    x_max_val = data_clean.max()

    dev_min = abs(mean_val - x_min_val)
    dev_max = abs(mean_val - x_max_val)

    if dev_max > dev_min:
        suspicious_val = x_max_val
        t_observed = dev_max / std_val
    else:
        suspicious_val = x_min_val
        t_observed = dev_min / std_val

    # Критическое значение Стьюдента
    df = n - 2
```

```
    t_critical = stats.t.ppf(p_outlier, df)

    print(f"\n--- Итерация {iteration} ---")
    print(f"n = {n}, M(x) = {mean_val:.2f}, S(x) = {std_val:.2f}")
    print(f"Подозрительное значение: {suspicious_val}")
    print(f"t_набл = {t_observed:.2f}")
    print(f"t_кр({p_outlier:.2f}; ν={df}) = {t_critical:.2f}")

    if t_observed > t_critical:
        print(f"Вывод: {t_observed:.2f} > {t_critical:.2f}, значение
      {suspicious_val} является выбросом и удаляется.")
        # Удаляем первое вхождение подозрительного значения
        data_clean = np.delete(data_clean, np.where(data_clean ==
      suspicious_val)[0][0])
        iteration += 1
    else:
        print(f"Вывод: {t_observed:.2f} <= {t_critical:.2f}, гипотеза об
      отсутствии выбросов принимается.")
        print("Процедура очистки завершена.")
        break

print(f"\nИтоговый объем очищенной выборки: n = {len(data_clean)}")
print("Очищенная выборка:")
print(np.sort(data_clean))

# --- 1.3. Определение доверительных интервалов ---
print("\n--- 1.3. Определение доверительных интервалов ---")

n_clean = len(data_clean)
mean_clean = np.mean(data_clean)
std_clean = np.std(data_clean, ddof=1)
p_conf = 0.95
alpha_conf = 1 - p_conf

print(f"Расчеты для очищенной выборки: n = {n_clean}, M(x) =
      {mean_clean:.2f}, S(x) = {std_clean:.2f}")

# 1. Доверительный интервал для математического ожидания
df_mean = n_clean - 1
# ppf для нахождения критической точки t для двустороннего интервала
      (alpha/2)
t_crit_mean = stats.t.ppf(1 - alpha_conf / 2, df_mean)
# Предельная погрешность (половина ширины интервала)
epsilon = t_crit_mean * (std_clean / np.sqrt(n_clean))

lower_bound_mean = mean_clean - epsilon
upper_bound_mean = mean_clean + epsilon

print("\n1. Доверительный интервал для математического ожидания:")
print(f"t_кр(α={alpha_conf:.2f}; ν={df_mean}) = {t_crit_mean:.3f}")
print(f"Предельная погрешность ε = {epsilon:.2f} HB")
print(f"Интервал: ({lower_bound_mean:.2f} HB; {upper_bound_mean:.2f} HB) с
      вероятностью {p_conf:.2f}")

# 2. Доверительный интервал для СКО
df_std = n_clean - 1
chi2_right = stats.chi2.ppf(1 - alpha_conf / 2, df_std)
chi2_left = stats.chi2.ppf(alpha_conf / 2, df_std)

lower_bound_std = std_clean * np.sqrt(df_std / chi2_right)
upper_bound_std = std_clean * np.sqrt(df_std / chi2_left)
```

```
print("\n2. Доверительный интервал для СКО:")
print(f"χ²_пр(α/2={alpha_conf/2:.3f}; ν={df_std}) = {chi2_right:.2f}")
print(f"χ²_лев(1-α/2={1-alpha_conf/2:.3f}; ν={df_std}) = {chi2_left:.2f}")
print(f"Интервал: ({lower_bound_std:.2f} HB; {upper_bound_std:.2f} HB) с
      вероятностью {p_conf:.2f}")

# --- 1.4. Расчет необходимого объема выборки ---
print("\n--- 1.4. Расчет необходимого объема выборки ---")

# Заданные параметры
epsilon_target = 10 # целевая погрешность
p_target = 0.99
alpha_target = 1 - p_target
# Используем S(x) из очищенной выборки как наилучшую оценку
s_for_calc = std_clean

print(f"Целевая погрешность ε = {epsilon_target} HB")
print(f"Доверительная вероятность p = {p_target}")
print(f"Оценка S(x) = {s_for_calc:.2f} HB")

# Итерационный поиск
n_min = 2
while True:
    df = n_min - 1
    # ppf для двустороннего интервала
    t_crit_target = stats.t.ppf(1 - alpha_target / 2, df)

    # Правая часть неравенства
    required_n_float = (s_for_calc * t_crit_target / epsilon_target)**2

    print(f"Проверка для n = {n_min}: t_кр = {t_crit_target:.3f}, требуемое n
      > {required_n_float:.2f}")

    if n_min > required_n_float:
        print(f"\nУсловие n > требуемого n выполнено. Минимальный объем
      выборки n_min = {n_min}")
        break

    n_min += 1
    if n_min > 1000: # Защита от бесконечного цикла
        print("Не удалось найти решение за 1000 итераций.")
        break

# --- Задача 2. Проверка настройки установки ---
print("\n--- Задача 2. Проверка настройки установки ---")

# 2.1. Исходные данные для Варианта 5
h_nominal = 50.0 # Номинальное значение M₀
tolerance = 1.0    # Допуск
data_task2 = np.array([
    50.7, 50.8, 50.6, 51.1, 49.4, 49.8, 50.7, 49.9, 50.1, 50.6,
    50.8, 52.0, 50.5, 49.7, 49.7, 51.6, 49.7, 50.5, 50.4, 50.1
])
n_task2 = len(data_task2)

print(f"Номинальная толщина M₀ = {h_nominal} мм")
print(f"Объем выборки n = {n_task2}")

# --- 2.2. Проверка гипотезы о равенстве среднего ---
print("\n--- 2.2. Проверка гипотезы о равенстве среднего ---")
```

```
alpha_task2 = 0.05

# Рассчитываем выборочные характеристики
mean_task2 = np.mean(data_task2)
std_task2 = np.std(data_task2, ddof=1) # ddof=1 для несмещенной оценки

print(f"Выборочное среднее M(h) = {mean_task2:.2f} мм")
print(f"Выборочное СКО S(h) = {std_task2:.2f} мм")

# Проводим одновыборочный t-тест
# Функция ttest_1samp из scipy.stats делает все вычисления автоматически
t_observed, p_value = stats.ttest_1samp(data_task2, h_nominal)

# Критическое значение для двустороннего теста
df_task2 = n_task2 - 1
t_critical = stats.t.ppf(1 - alpha_task2 / 2, df_task2)

print(f"\nРезультаты t-теста (уровень значимости α = {alpha_task2}):")
print(f"t_набл = {abs(t_observed):.2f}")
print(f"t_кр(α={alpha_task2}; ν={df_task2}) = {t_critical:.3f}")
print(f"P-value = {p_value:.4f}")

# Вывод
if abs(t_observed) > t_critical:
    print(f"\nВывод: Так как t_набл > t_кр, нулевая гипотеза H₀ отвергается.")
    print("Расхождение является статистически значимым. Установка требует
      регулировки.")
else:
    print(f"\nВывод: Так как t_набл <= t_кр, нулевая гипотеза H₀ не
      отвергается.")
    print("Нет оснований считать, что установка настроена неверно.")

# --- Задача 3. Сравнительная оценка эффективности ПО ---
print("\n\n--- Задача 3. Сравнительная оценка эффективности ПО ---")

# 3.1. Исходные данные для Варианта 5 (Б, Исп. 2, Исп. 4)
# Объединяем данные из трех лабораторий для каждого типа ПО

data_b = np.array([
    8, 11, 10, 9, 8, 8, 10, 13, 9, 7, 9, 9,  # Лаб 1
    11, 11, 12, 11, 7, 8, 8, 11, 9, 9, 11, 9,  # Лаб 2
    10, 8, 12, 10, 10, 12, 9, 11, 12, 8, 13, 9 # Лаб 3
])

data_isp2 = np.array([
    11, 15, 11, 10, 10, 12, 11, 11, 11, 12, 14, 12, # Лаб 1
    13, 9, 9, 12, 10, 9, 11, 10, 12, 11, 12, 11,   # Лаб 2
    10, 11, 10, 12, 13, 9, 9, 13, 11, 11, 10, 10  # Лаб 3
])

data_isp4 = np.array([
    12, 11, 13, 9, 14, 12, 9, 12, 13, 15, 15, 14, # Лаб 1
    14, 13, 13, 10, 13, 13, 12, 9, 15, 13, 11, 15, # Лаб 2
    13, 15, 9, 11, 10, 12, 11, 14, 8, 11, 12, 11  # Лаб 3
])

print(f"Объем выборок: n_Б = {len(data_b)}, n_Исп2 = {len(data_isp2)}, n_Исп4
      = {len(data_isp4)}")

# --- 3.2.1. Сравнение: Базовое ПО vs Исполнение 2 ---
print("\n--- 3.2.1. Сравнение: Базовое (Б) vs Исполнение 2 ---")
```

```
alpha_task3 = 0.05

# 1. Предварительные статистики
mean_b = np.mean(data_b)
std_b = np.std(data_b, ddof=1)
mean_isp2 = np.mean(data_isp2)
std_isp2 = np.std(data_isp2, ddof=1)
print(f"Статистики Б: M(x) = {mean_b:.2f}, S(x) = {std_b:.2f}")
print(f"Статистики Исп. 2: M(x) = {mean_isp2:.2f}, S(x) = {std_isp2:.2f}")

# 2. F-тест на равенство дисперсий
# Убедимся, что в числителе большая дисперсия
if std_b**2 > std_isp2**2:
    f_observed = std_b**2 / std_isp2**2
    dfn, dfd = len(data_b) - 1, len(data_isp2) - 1
else:
    f_observed = std_isp2**2 / std_b**2
    dfn, dfd = len(data_isp2) - 1, len(data_b) - 1

f_critical = stats.f.ppf(1 - alpha_task3, dfn, dfd)
print(f"\nF-тест: F_набл = {f_observed:.2f}, F_кр = {f_critical:.2f}")
if f_observed > f_critical:
    print("Вывод: Дисперсии НЕ равны.")
    equal_var_flag = False
else:
    print("Вывод: Дисперсии можно считать равными.")
    equal_var_flag = True

# 3. T-тест для независимых выборок
# alternative='less' проверяет гипотезу M(a) < M(b)
t_observed, p_value = stats.ttest_ind(data_b, data_isp2,
      equal_var=equal_var_flag, alternative='less')
df_t = len(data_b) + len(data_isp2) - 2
t_critical = stats.t.ppf(alpha_task3, df_t) # Критическая точка для
      левостороннего теста

print(f"\nT-тест: t_набл = {t_observed:.2f}, t_кр(одностор.) =
      {t_critical:.2f}, p-value = {p_value:.4f}")
if t_observed < t_critical:
    print("Вывод: H0 отвергается. Эффективность Исп. 2 выше базовой.")
else:
    print("Вывод: H0 не отвергается. Нет значимой разницы в эффективности.")

# --- 3.2.2. Сравнение: Базовое ПО vs Исполнение 4 ---
print("\n--- 3.2.2. Сравнение: Базовое (Б) vs Исполнение 4 ---")
mean_isp4 = np.mean(data_isp4)
std_isp4 = np.std(data_isp4, ddof=1)
print(f"Статистики Б: M(x) = {mean_b:.2f}, S(x) = {std_b:.2f}")
print(f"Статистики Исп. 4: M(x) = {mean_isp4:.2f}, S(x) = {std_isp4:.2f}")

# F-тест
f_observed = std_isp4**2 / std_b**2
dfn, dfd = len(data_isp4) - 1, len(data_b) - 1
f_critical = stats.f.ppf(1 - alpha_task3, dfn, dfd)
print(f"\nF-тест: F_набл = {f_observed:.2f}, F_кр = {f_critical:.2f}")
if f_observed > f_critical:
    print("Вывод: Дисперсии НЕ равны.")
    equal_var_flag = False
else:
    print("Вывод: Дисперсии можно считать равными.")
    equal_var_flag = True
```

```
# T-тест
t_observed, p_value = stats.ttest_ind(data_b, data_isp4,
      equal_var=equal_var_flag, alternative='less')
df_t = len(data_b) + len(data_isp4) - 2
t_critical = stats.t.ppf(alpha_task3, df_t)
print(f"\nT-тест: t_набл = {t_observed:.2f}, t_кр(одностор.) =
      {t_critical:.2f}, p-value = {p_value:.8f}")
if t_observed < t_critical:
    print("Вывод: H0 отвергается. Эффективность Исп. 4 выше базовой.")
else:
    print("Вывод: H0 не отвергается.")

# --- 3.2.3. Сравнение: Исполнение 2 vs Исполнение 4 ---
print("\n--- 3.2.3. Сравнение: Исполнение 2 vs Исполнение 4 ---")
print(f"Статистики Исп. 2: M(x) = {mean_isp2:.2f}, S(x) = {std_isp2:.2f}")
print(f"Статистики Исп. 4: M(x) = {mean_isp4:.2f}, S(x) = {std_isp4:.2f}")

# F-тест
f_observed = std_isp4**2 / std_isp2**2
dfn, dfd = len(data_isp4) - 1, len(data_isp2) - 1
f_critical = stats.f.ppf(1 - alpha_task3, dfn, dfd)
print(f"\nF-тест: F_набл = {f_observed:.2f}, F_кр = {f_critical:.2f}")
if f_observed > f_critical:
    print("Вывод: Дисперсии НЕ равны.")
    equal_var_flag = False
else:
    print("Вывод: Дисперсии можно считать равными.")
    equal_var_flag = True

# T-тест (двусторонний, т.к. заранее не знаем, что лучше)
# equal_var=False автоматически включает поправку Уэлча
t_observed, p_value = stats.ttest_ind(data_isp2, data_isp4,
      equal_var=equal_var_flag, alternative='two-sided')
# При неравных дисперсиях df считается по сложной формуле, но scipy делает
      это за нас
# Для отчета можно привести примерное значение, либо взять из p-value
df_welch_approx = 65 # Примерное значение для отчета
t_critical = stats.t.ppf(1 - alpha_task3 / 2, df_welch_approx)
print(f"\nT-тест (Уэлча): t_набл = {t_observed:.2f}, t_кр(двустор.) ≈
      ±{t_critical:.2f}, p-value = {p_value:.4f}")
if p_value < alpha_task3:
    print("Вывод: H0 отвергается. Есть разница в эффективности.")
else:
    print("Вывод: H0 не отвергается.")
```