

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФГБОУ ВО «Воронежский государственный
технический университет»**

**Кафедра автоматизированного оборудования
машиностроительного производства**

**КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКАЯ
ИНФОРМАТИКА**

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению курсовой работы для студентов направления
15.03.05 «Конструкторско-технологическое обеспечение
машиностроительных производств» (профили «Технология
машиностроения», «Металлообрабатывающие станки
и комплексы», «Конструкторско-технологическое
обеспечение кузнечно-штамповочного производства»)
всех форм обучения

Воронеж 2021

УДК 621.01(07)
ББК 34.5я7

Составитель ст. преп. С. Л. Новокшенов

Конструкторско-технологическая информатика: методические указания к выполнению курсовой работы для студентов направления 15.03.05 «Конструкторско-технологическое обеспечение машиностроительных производств» (профили «Технология машиностроения», «Металлообрабатывающие станки и комплексы», «Конструкторско-технологическое обеспечение кузнечно-штамповочного производства») всех форм обучения / ФГБОУ ВО «Воронежский государственный технический университет»; сост.: С. Л. Новокшенов. – Воронеж: Изд-во ВГТУ, 2021. – 28 с.

В методических указаниях изложены основные положения курса, приведены необходимые теоретические сведения для выполнения курсовой работы, приведен список литературы.

Предназначены для студентов направления 15.03.05 «Конструкторско-технологическое обеспечение машиностроительных производств» (профили «Технология машиностроения», «Металлообрабатывающие станки и комплексы», «Конструкторско-технологическое обеспечение кузнечно-штамповочного производства») всех форм обучения.

Методические указания подготовлены в электронном виде и содержатся в файле МУ_КР_КТИ.pdf

Ил. 21. Библиогр.: 6 назв.

УДК 621.01(07)
ББК 34.5я7

Рецензент – А. В. Демидов, канд. техн. наук, доц.
кафедры автоматизированного
оборудования машиностроительного
производства ВГТУ

*Издается по решению редакционно-издательского совета
Воронежского государственного технического университета*

ВВЕДЕНИЕ

Выполнение курсовой работы по дисциплине «Конструкторско-технологическая информатика» позволит студентам закрепить навыки, полученные при выполнении лабораторных работ.

Целью выполнения курсовой работы является закрепление и развитие практических навыков разработки прикладных программ на языке программирования Python.

В курсовой работе обучающиеся используют материалы из справочной литературы, ГОСТов, повышают навыки работы в стандартных программных средах.

Выполнение курсовой работы в соответствии с заданием и выполнением требований, указанных во внутреннем документе ФГБОУ ВО ВГТУ «Требования к оформлению курсовых проектов и работ по программам высшего образования» позволяет студенту полностью освоить компетенцию

1. ПОСТАНОВКА ЗАДАЧИ

После получения темы работы необходимо выполнить анализ задачи, т. е. определить входные, выходные и управляющие параметры изучаемого процесса и их взаимосвязь.

Рассмотрим примеры и подходы к выполнению заданий, которые можно применять при выполнении курсовой работы.

Пример 1. Разработка линейного вычислительного алгоритма. Необходимо написать программу, выполняющую вычисление массы поковки детали по формуле:

$$G_{\text{п}} = 1,25 \cdot G_{\text{д}}, \quad (1)$$

где $G_{\text{п}}$, кг – масса поковки;

$G_{\text{д}}$, кг – масса детали.

Пример 2. Применение условного оператора. Задача – вычислить диаметр заготовки по формуле (2)

$$D'_{\text{ЗАГ}} = 1,08 \cdot \sqrt[3]{\frac{V'_{\text{ЗАГ}}}{m}}, \quad (2)$$

где $D'_{\text{ЗАГ}}$, мм – диаметр заготовки;

$V'_{\text{ЗАГ}}$, мм³ – объем заготовки;

m – коэффициент отношения диаметра заготовки к высоте (1,7...2,5).

Пример 3. Вычисления с циклом. Задача – вычислить диаметр заготовки по формуле (2) для нескольких значений коэффициента m .

Пример 4. Разработка функции. Вычислить диаметр заготовки по формуле в примере 3 с использованием функций в Python.

Пример 5. Работа со строками. Задача – оформить вывод сообщений программы в консоли с применением строк и операторов работами со строками.

2. РЕШЕНИЕ ЗАДАЧИ

Пример 1 Разработка линейного вычислительного алгоритма. Исходными данными в данном случае является G_d , кг, подставив значение которого в формулу, получим искомый результат.

Пример 2. Применение условного оператора. Исходными данными в рассматриваемом случае будут значения коэффициента m и объема заготовки, которые должны быть введены с пользователем.

Пример 3. Вычисления с циклом. Исходными данными в рассматриваемом случае будут значения коэффициента m и объема заготовки, которые должны быть введены с пользователем. Далее с помощью цикла идет перебор значений коэффициента m с шагом 0,5 от введенного значения, но не больше 2,5 (предела, после превышения которого наступит потеря устойчивости заготовки).

Пример 4. Разработка функции. Исходными данными в рассматриваемом случае будут значения коэффициента m и объема заготовки, которые должны быть введены с пользователем. Далее с помощью цикла идет перебор значений коэффициента m с шагом 0,5 от введенного значения, но не больше 2,5 (предела, после превышения которого наступает потеря устойчивости заготовки).

Отличием от примера 3 является оформление вычислений в виде функции, которая обозначается служебным словом **def:** (рис.) на языке программирования Python

В качестве аргумента в разрабатываемой функции используется значение коэффициента отношения длины к диаметру m .

```
def Diamzag(m):
    try:
        Dzag=1.08*(Vzag/m)**(1/3)
        print('m = '+str(m)+' Dzag = '+str(Dzag))
    #выводим значение m и рассчитанного диаметра
    except ZeroDivisionError:
```

```

print('Деление на ноль!')
#вывод в консоль сообщения о делении на ноль
m=float (input("Введите значение коэффициента отношения L
к D заготовки, отличное от нуля = "))

```

Рис. . – Текст функции с аргументом

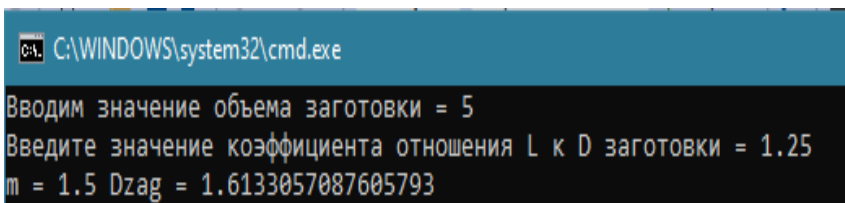
Пример 5. Работа со строками. Используя функции преобразования численных типов в строку `str()` и операторы '+'. Данные одного строкового типа в операторе `print` можно объединять в одну строку, что позволяет создать сообщения программы, понятные пользователю:

```

print('m = '+str(m)+' Dzag = '+str(Dzag)) #выводим значение m
и рассчитанного диаметра

```

а)



б)

Рис. 1. Текст оператора и пример работы

3. СОЗДАНИЕ АЛГОРИТМА ПРОГРАММЫ

Прежде, чем решать какую-либо задачу с применением ЭВМ, необходимо разработать алгоритм её решения с учётом требований и правил теории алгоритмов. Вместе с математической логикой теория алгоритмов образует теоретическую основу вычислительных наук.

Алгоритмы удобнее всего представлять графически в виде блок-схем, для чего в том же Microsoft Word есть специальный инструмент – блок-схемы, который находится в фигурах и в котором приведены все основные необходимые элементы (рис. 2).

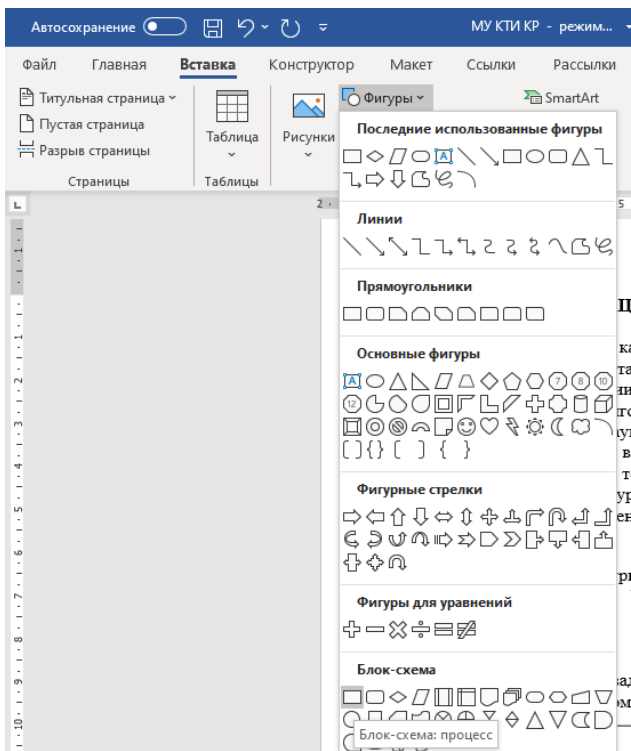


Рис. 2. Инструмент Фигуры текстового редактора Microsoft Word

Пример 1. Разработка линейного вычислительного алгоритма. Алгоритм решения задачи в данном случае будет выглядеть следующим образом (рис. 3).

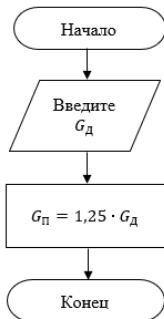


Рис. 3. Алгоритм решения задачи вычисления массы
ПОВОККИ

Пример 2. Применение условного оператора.

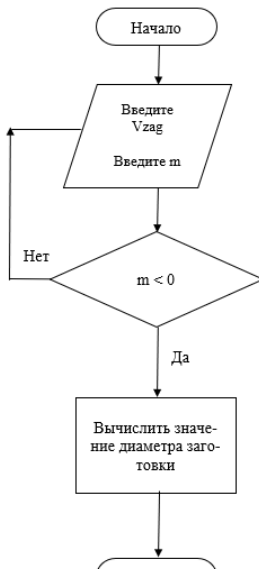


Рис. 4. Алгоритм решения задачи вычисления диаметра
заготовки

Во всех подобных случаях структура алгоритма будет аналогичной, т. е. линейной. В случае использования логических операторов и циклов структура алгоритма будет иной, как показано в других примерах.

Пример 3. Вычисления с циклом

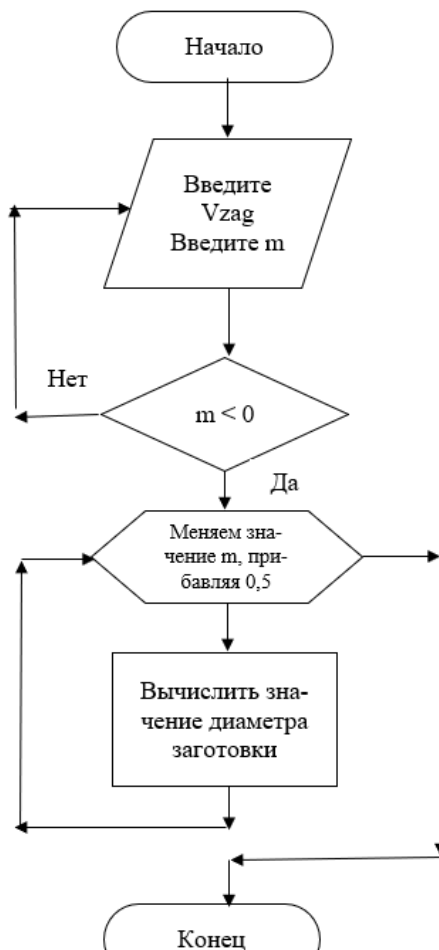


Рис. 5. Алгоритм решения задачи вычисления диаметра заготовки с циклом

Пример 4. Разработка функции. Алгоритм аналогичен алгоритму в примере 3, только вычислительный блок оформлен в виде функции, что позволяет её использовать многократно вызовом в соответствующем месте в программе (рис. 6).

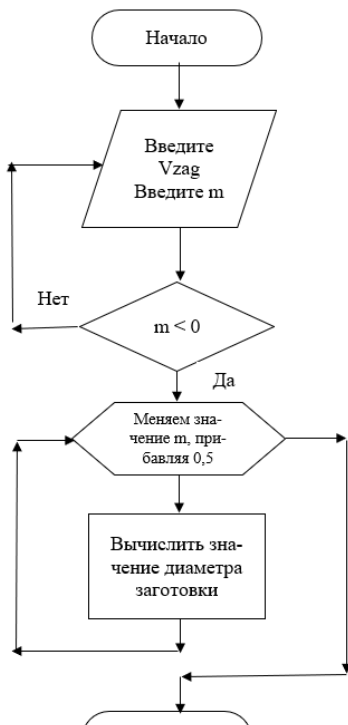


Рис. 6. Алгоритм решения задачи вычисления диаметра заготовки с функцией

Пример 5. Работа со строками. Алгоритм программы аналогичен предыдущим, разница в выводе текстовых сообщений.

Пример 6. Создание фронт-энда программы. Алгоритм программы аналогичен предыдущим, разница в структуре текста программы и подходах к реализации.

4. РЕАЛИЗАЦИЯ АЛГОРИТМА НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ

Последовательность действий при решении задач всегда одинакова при выборе в качестве IDE Microsoft Visual Studio.Community.

Вначале создается новый проект (рис. 7).

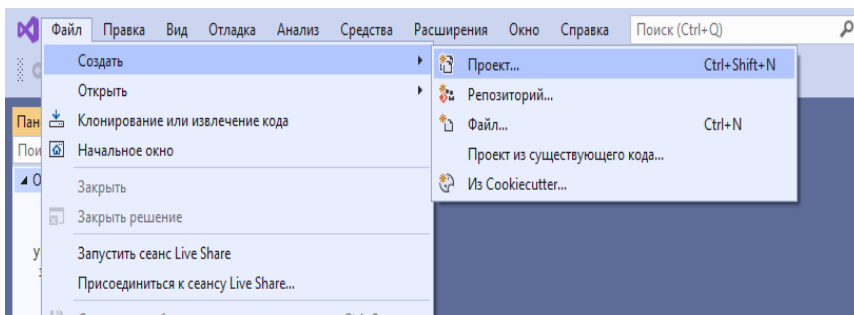


Рис. 7. Создание нового проекта

Далее выбирается язык программирования, на котором будет реализовываться разработанный алгоритм (рис. 8).

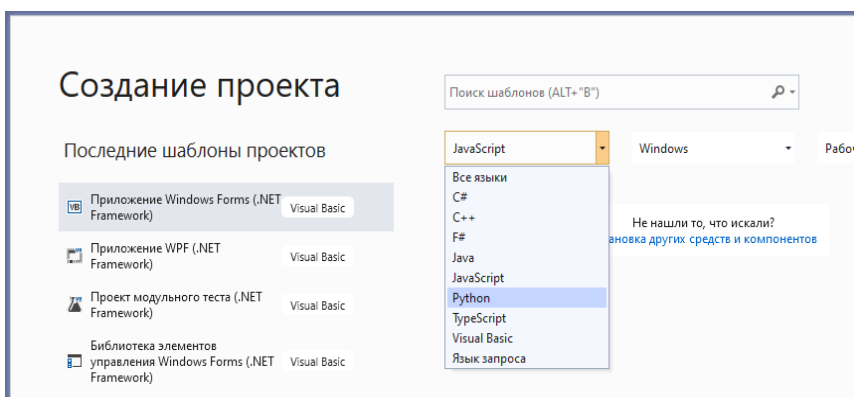


Рис. 8. Выбор языка программирования

Затем осуществляется выбор среды, в которой будет работать программа и в которой будут отображаться результаты вычислений. Для первых программ рекомендуется создавать

консольное приложение, т. е. программу, работающую в командной строке (рис. 9).

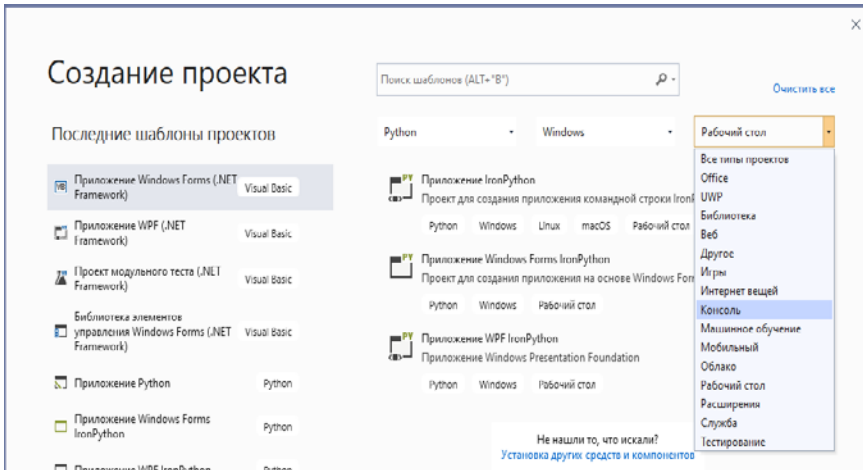


Рис. 9. Выбор среды выполнения программы (консоль)

Далее останется выбрать Приложение Python и войти в текстовый редактор для набора кода по щелчку по кнопке.

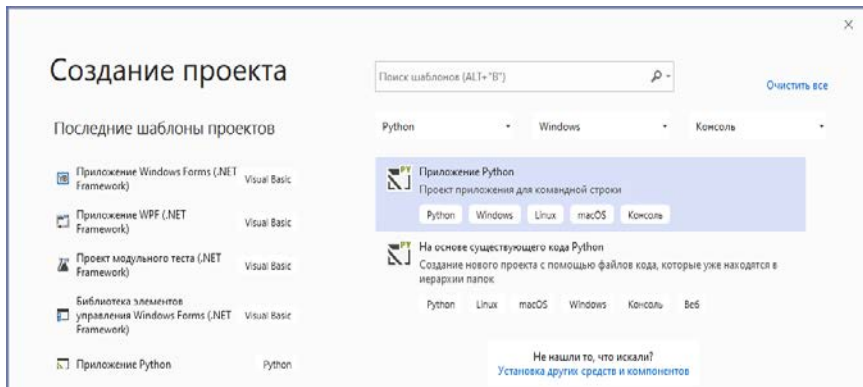
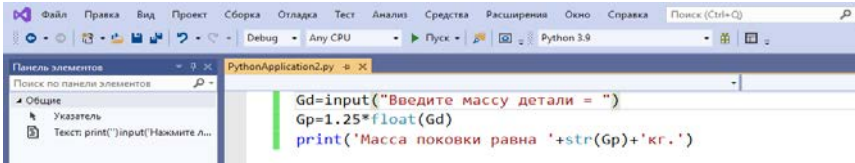


Рис. 10. Выбор параметров проекта в диалоге «Создание проекта»

Текст программы показан на рис. 11.

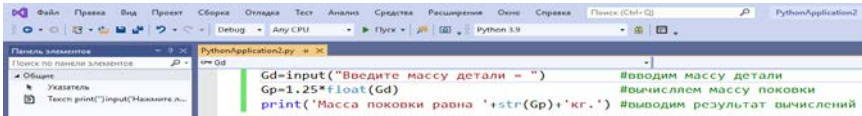
Пример 1



```
Gd=input("Введите массу детали = ")
Gp=1.25*float(Gd)
print('Масса поковки равна '+str(Gp)+'кг.')
```

Рис. 11. Текст программы без комментариев

Хорошим тоном при программировании считается оставлять подробные пояснения в виде *комментариев*. Под комментарием понимается текстовая строка, которая не будет восприниматься компилятором как часть программного кода. Для языка Python комментарии вводятся после символа #.



```
Gd=input("Введите массу детали = ") #вводим массу детали
Gp=1.25*float(Gd) #вычисляем массу поковки
print('Масса поковки равна '+str(Gp)+'кг.') #выводим результат вычислений
```

Рис. 12. Текст программы с комментариями

Пример 2

```
Vzag=float (input("Вводим значение объема заготовки = ") )
#вводим значение Vzag
m=float (input("Введите значение коэффициента отношения L к D
заготовки = ")) #вводим значение m
try:
    Dzag=1.08*(Vzag/m)**(1/3)
#вычисляем значение Dzag
    print(Dzag)
#выводим рассчитанное значение Dzag в консоль
except ZeroDivisionError:
    print('Деление на ноль!')
#вывод в консоль сообщения о делении на ноль
m=float (input("Введите значение коэффициента отношения L
к D заготовки, отличное от нуля = "))
Dzag=1.08*(Vzag/m)**(1/3)
#ещё раз вычисляем значение Dzag
print(Dzag)
#вычисляем значение Dzag
```

Рис. 13. Текст программы с комментариями

Пример 3

```
Vzag=float (input("Вводим значение объема заготовки = ") )
#вводим значение Vzag
m=float (input("Введите значение коэффициента отношения L к D
заготовки = ")) #вводим значение m
try:
    Dzag=1.08*(Vzag/m)**(1/3)
#вычисляем значение Dzag
    print(Dzag)
#выводим рассчитанное значение Dzag в консоль
except ZeroDivisionError:
    print('Деление на ноль!')
#вывод в консоль сообщения о делении на ноль
    m=float (input("Введите значение коэффициента отношения L
к D заготовки, отличное от нуля = "))
    Dzag=1.08*(Vzag/m)**(1/3)
#ещё раз вычисляем значение Dzag
    print(Dzag)
#вычисляем значение Dzag
```

Рис. 14. Текст программы

Пример 4

```
Vzag=float (input("Вводим значение объема заготовки = ") )
#вводим значение Vzag
m=float (input("Введите значение коэффициента отношения L к D
заготовки = ")) #вводим значение m

def Diamzag(m):
    try:
        Dzag=1.08*(Vzag/m)**(1/3)
        print('m = '+str(m)+' Dzag = '+str(Dzag))
#выводим значение m и рассчитанного диаметра
    except ZeroDivisionError:
        print('Деление на ноль!')
#вывод в консоль сообщения о делении на ноль
        m=float (input("Введите значение коэффициента отношения L
к D заготовки, отличное от нуля = "))

try:
    Dzag=1.08*(Vzag/m)**(1/3)
    while m < 2.5:
        m=m+0.25
        Diamzag(m)
```

```

except ZeroDivisionError:
    print('Деление на ноль!')
#вывод в консоль сообщения о делении на ноль
m=float (input("Введите значение коэффициента отношения L
к D заготовки, отличное от нуля = "))
while m < 2.5:
    m=m+0.25
    Diamzag(m)

```

Рис. 15.Текст программы

Таким образом, однократное объявление функции позволяет использовать её многократно, что позволяет сократить число одинаковых строк в программе и улучшить её читабельность.

Пример 4.1

```

def get_temp_comfort(temp):
#создание словаря с выбором по значению
    return {
        temp <= -20: 'холодно',
        -20 <= temp <= 0: 'прохладно',
        0 <= temp <= 15: 'зябко',
        15 <= temp <= 25: 'тепло',
        25 <= temp: 'жарко'
    }[True]

temp=float (input("Введите значение температуры окружающего
воздуха, T = ")) #Ввод значения температуры
if (temp >= -20 and temp <= 25):
#Проверка вхождения значения в допустимый интервал
    print("В производственных помещениях:
",get_temp_comfort(temp))
else:
    print("Значение температуры выходит за допустимые преде-
лы!")

```

Рис. 16

5. РЕЗУЛЬТАТЫ РАБОТЫ ПРОГРАММ

Результат вычислений будет выведен в команду строку (рис. 17).

Пример 1

```
C:\WINDOWS\system32\cmd.exe
Введите массу детали = 5
Масса поковки равна 6.25кг.
Для продолжения нажмите любую клавишу . . .
```

а)

```
C:\WINDOWS\system32\cmd.exe
Введите массу детали = 8
Масса поковки равна 10.0кг.
Для продолжения нажмите любую клавишу . . .
```

б)

Рис. 17. Результат вычислений

Пример 2

```
C:\WINDOWS\system32\cmd.exe
Вводим значение объема заготовки = 100000
Введите значение коэффициента отношения L к D заготовки = 1.25
46.53578930468868
Для продолжения нажмите любую клавишу . . .
```

а)

```
C:\WINDOWS\system32\cmd.exe
Вводим значение объема заготовки = 100000
Введите значение коэффициента отношения L к D заготовки = 0
Деление на ноль!
Введите значение коэффициента отношения L к D заготовки, отличное от нуля = 1.25
46.53578930468868
Для продолжения нажмите любую клавишу . . .
```

б)

Рис. 18. Результат вычислений

Пример 3

```
C:\WINDOWS\system32\cmd.exe
Вводим значение объема заготовки = 100000
Введите значение коэффициента отношения L к D заготовки = 0
Деление на ноль!
Введите значение коэффициента отношения L к D заготовки, отличное от нуля = _
```

а)

```
C:\WINDOWS\system32\cmd.exe
Вводим значение объема заготовки = 100000
Введите значение коэффициента отношения L к D заготовки = 0
Деление на ноль!
Введите значение коэффициента отношения L к D заготовки, отличное от нуля = 1.2
m = 1.45 Dzag = 44.28953168209481
m = 1.7 Dzag = 42.00240082314458
m = 1.95 Dzag = 40.1247388065262
m = 2.2 Dzag = 38.54335619719483
m = 2.45 Dzag = 37.1850520758464
m = 2.7 Dzag = 36.0
Для продолжения нажмите любую клавишу . . .
```

б)

Рис. 19. Результат вычислений

Пример 4

Результат работы программы будет аналогичен примеру 3, вся разница в тексте самой программы (рис. 20).

```
C:\WINDOWS\system32\cmd.exe
Вводим значение объема заготовки = 100000
Введите значение коэффициента отношения L к D заготовки = 0
Деление на ноль!
Введите значение коэффициента отношения L к D заготовки, отличное от нуля = _
```

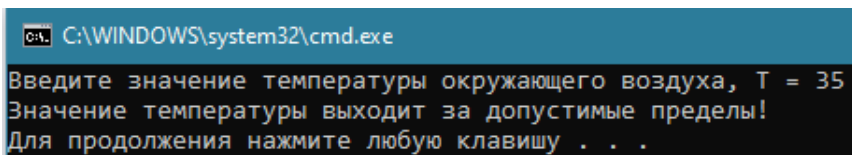
а)

```
C:\WINDOWS\system32\cmd.exe
Вводим значение объема заготовки = 100000
Введите значение коэффициента отношения L к D заготовки = 0
Деление на ноль!
Введите значение коэффициента отношения L к D заготовки, отличное от нуля = 1.2
m = 1.45 Dzag = 44.28953168209481
m = 1.7 Dzag = 42.00240082314458
m = 1.95 Dzag = 40.1247388065262
m = 2.2 Dzag = 38.54335619719483
m = 2.45 Dzag = 37.1850520758464
m = 2.7 Dzag = 36.0
Для продолжения нажмите любую клавишу . . .
```

б)

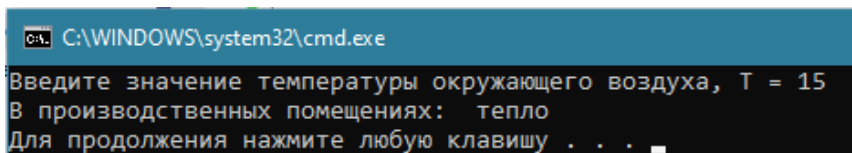
Рис. 20. Результат вычислений

Пример 4.1



```
C:\WINDOWS\system32\cmd.exe
Введите значение температуры окружающего воздуха, T = 35
Значение температуры выходит за допустимые пределы!
Для продолжения нажмите любую клавишу . . .
```

а)



```
C:\WINDOWS\system32\cmd.exe
Введите значение температуры окружающего воздуха, T = 15
В производственных помещениях: тепло
Для продолжения нажмите любую клавишу . . .
```

б)

Рис. 21. Результат определения комфортности температуры

ТЕМЫ КУРСОВОЙ РАБОТЫ

БКТ0-_01 (технология машиностроения), БКТ0-_02 («Технология машиностроения», «Машиностроение» и «Металлообрабатывающие станки и комплексы» очной формы обучения), а также студентов профиля «Машиностроения» заочной формы обучения

- 1) Автоматизация расчета скорости резания при продольном точении ([2], стр. 265)
- 2) Автоматизация расчета скорости резания при поперечном точении ([2], стр. 265)
- 3) Автоматизация расчета скорости резания при отрезании ([2], стр. 292)
- 4) Автоматизация расчета скорости резания при нарезании резьбы ([2], стр. 293)
- 5) Автоматизация расчета скорости резания при фасонном точении ([2], стр. 265)
- 6) Автоматизация расчета силы резания при продольном точении ([2], стр. 265)
- 7) Автоматизация расчета силы скорости и мощности резания при поперечном точении ([2], стр. 265)
- 8) Автоматизация расчета силы скорости и мощности резания при отрезании ([2], стр. 265)
- 9) Автоматизация расчета силы и мощности резания при нарезании ([2], стр. 293)
- 10) Автоматизация расчета силы и мощности резания при фасонном точении ([2], стр. 265)
- 11) Автоматизация расчета скорости резания при сверлении ([2], стр. 276)
- 12) Автоматизация расчета скорости резания при рассверливании ([2], стр. 276)
- 13) Автоматизация расчета скорости резания при зенкеро-вании ([2], стр. 276)
- 14) Автоматизация расчета скорости резания при развертывании ([2], стр. 276)

15) Автоматизация расчета крутящего момента и мощности резания при разворачивании ([2], стр. 276)

16) Автоматизация расчета подачи и скорости резания при фрезеровании ([2], стр. 281)

17) Автоматизация расчета крутящего момента и мощности резания при фрезеровании ([2], стр. 265)

18) Автоматизация расчета скорости резания при нарезании крепежной резьбы резцами с пластинами из твердого сплава ([2], стр. 293)

19) Автоматизация расчета скорости резания при нарезании крепежной резьбы резцами из быстрорежущей стали ([2], стр. 293)

20) Автоматизация расчета скорости резания при вихревом нарезании метрической и трапецеидальной резьб твердосплавными резцами во вращающихся головках ([2], стр. 293)

21) Автоматизация расчета скорости резания при нарезании резьбы с ограниченным выходом резца ([2], стр. 293)

22) Автоматизация расчета тангенциальной составляющей силы резания при нарезании резьбы резцами ([2], стр. 293)

23) Автоматизация расчета крутящего момента при нарезании резьбы метчиками ([2], стр. 293)

24) Автоматизация расчета крутящего момента при нарезании резьбы резьбовыми головками ([2], стр. 293)

25) Автоматизация расчета мощности при нарезании резьбы резцами ([2], стр. 293)

26) Автоматизация расчета мощности при нарезании резьбы метчиками, плашками и резьбовыми головками ([2], стр. 293)

27) Автоматизация расчета мощности при вихревом нарезании треугольной резьбы ([2], стр. 265)

28) Автоматизация расчета мощности при вихревом нарезании трапецеидальной резьбы ([2], стр. 265)

БКТ0-_03 (профиль «Конструкторско-технологическое обеспечение кузнечно-штамповочного производства»)

- 1) Автоматизация расчета степени сложности поковки ([4], стр. 7)
- 2) Автоматизация расчета объема заготовки на одну поковку ([4], стр. 75)
- 3) Автоматизация расчета длины исходной заготовки на одну поковку ([4], стр. 227)
- 4) Автоматизация расчета диаметра заготовки при закрытой штамповке ([4], стр. 79)
- 5) Автоматизация расчета массы поковки ([4], стр. 81)
- 6) Автоматизация расчета массы заготовки с учетом отходов на облой и угар ([4], стр. 82)
- 7) Автоматизация расчета объема поперечного заусенца ([4], стр. 196)
- 8) Автоматизация расчета усилия при штамповке осаживанием в открытых штампах для круглых и квадратных в плане поковок ([4], стр. 201)
- 9) Автоматизация расчета усилия для удлиненных в плане поковок ([4], стр. 201)
- 10) Автоматизация расчета при штамповке осаживанием в закрытых штампах ([4], стр. 202)
- 11) Автоматизация расчета деформирующего усилия вытяжки на гидравлическом прессе ([4], стр. 222)
- 12) Автоматизация расчета деформирующего усилия при штамповке на винтовых прессах ([4], стр. 234)
- 13) Автоматизация расчета усилия горизонтальноковочной машины при штамповке в закрытых формовочных и прошивных ручьях ([4], стр. 258)
- 14) Автоматизация обоснования применения заготовки на штамповку двух поковок ([4], стр. 292)
- 15) Автоматизация расчета усилия гибки V-образных деталей ([4], стр. 339)
- 16) Автоматизация расчета усилия гибки П-образных деталей ([4], стр. 340)
- 17) Автоматизация расчета усилия сдвига при отрезке при угле пересечения, равным нулю ([6], стр. 31)

- 18) Автоматизация расчета усилия сдвига при отрезке при угле пересечения при пересечении кромок под углом φ (т. [6], стр. 31)
- 19) Автоматизация расчета работы сдвига при отрезке (т. [5], стр. 32)
- 20) Автоматизация расчета усилия гибки в штампе ([6]стр. 94)
- 21) Автоматизация расчета усилия прижима при вытяжке ([6], стр. 126)
- 22) Автоматизация расчета работы деформации при вытяжке ([6], стр. 126)
- 23) Автоматизация расчета максимального усилия вытяжки для второй и последующей операций ([6], стр. 144)
- 24) Автоматизация расчета усилия прижима при вытяжке ([6], стр. 145)
- 25) Автоматизация расчета работы формоизменения при вытяжке ([6], стр. 145)

ТЕОРЕТИЧЕСКИЕ ВОПРОСЫ

БКТО- _01 (технология машиностроения), БКТО- _02 («Технология машиностроения», «Машиностроение» и «Металлообрабатывающие станки и комплексы» очной формы обучения), а также студентов профиля «Машиностроения» заочной формы обучения

- 1) Станки токарные: технологические возможности и область применения;
- 2) Станки фрезерные: технологические возможности и область применения;
- 3) Станки сверлильные: технологические возможности и область применения;
- 4) Станки шлифовальные: технологические возможности и область применения;
- 5) Станки
- 6) Процессы резания при продольном точении
- 7) Процессы резания при поперечном точении
- 8) Процессы отрезания заготовок
- 9) Особенности процессов нарезания резьбы
- 10) Процессы резания при фасонном точении
- 11) Процессы резания при сверлении
- 12) Процессы резания при рассверливании
- 13) Процессы резания при зенкеровании
- 14) Процессы резания при развертывании
- 15) Процессы резания при фрезеровании
- 16) Способы и инструмент нарезания крепежной резьбы
- 17) Особенности процесса нарезания крепежной резьбы резцами

БКТО-_03 (профиль «Конструкторско-технологическое обеспечение кузнечно-штамповочного производства»)

- 1) Прессы механические: технологические возможности и область применения;
- 2) Молоты штамповочные: технологические возможности и область применения;
- 3) Прессы гидравлические: технологические возможности и область применения;
- 4) Горизонтально-ковочные машины: технологические возможности и область применения;
- 5) Классификация и понятие степени сложности поковок
- 6) Особенности расчета параметров заготовки при открытой штамповке
- 7) Особенности расчета параметров заготовки при закрытой штамповке
- 8) Особенности расчета массы поковки
- 9) Особенности расчета массы поковки при открытой штамповки
- 10) Особенности расчета усилия при штамповке осаживанием в открытых штампах
- 11) Особенности расчета усилия при штамповке осаживанием в закрытых штампах
- 12) Особенности расчета деформирующего усилия вытяжки на гидравлическом прессе
- 13) Особенности технологических процессов на штамповку двух поковок
- 14) Особенности процессов сдвига при отрезке
- 15) Особенности процессов вытяжки

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Справочник технолога-машиностроителя. В 2-х т. Т. 1/Под ред. А. Г. Косиловой и Р. К. Мещерякова. – 4-е изд., перераб. и доп. – М.: Машиностроение, 1986, 656 с.

2. Справочник технолога-машиностроителя. В 2-х т. Т. 2/Под ред. А. Г. Косиловой и Р. К. Мещерякова. – 4-е изд., перераб. и доп. – М.: Машиностроение, 1985, 496 с.

3. Ковка и штамповка: Справочник. В 4-х т./Ред. совет: Е. И. Семенов (пред.) и др. – М.: Машиностроение, 1985 – т. 1, Материалы и нагрев. Оборудование. Ковка/Под ред. Е. И. Семенова, 1985. 568 с.

4. Ковка и штамповка: Справочник. В 4-х т./Ред. совет: Е. И. Семенов (пред.) и др. – М.: Машиностроение, 1985 – т. 2, Горячая штамповка. Ковка/Под ред. Е. И. Семенова, 1986, 592 с.

5. Ковка и штамповка: Справочник. В 4-х т./Ред. совет: Е. И. Семенов (пред.) и др. – М.: Машиностроение, 1985 – т. 3, Холодная объемная штамповка /Под ред. Е. И. Семенова, 1987, 384 с.

6. Ковка и штамповка: Справочник. В 4-х т./Ред. совет: Е. И. Семенов (пред.) и др. – М.: Машиностроение, 1985 – т. 4, Листовая штамповка./Под ред. Е. И. Семенова, 1985-1987. 544 с.

ПРИЛОЖЕНИЕ

Титульный лист курсовой работы*

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГТУ»)

Факультет машиностроения и аэрокосмической техники
Кафедра автоматизированного оборудования машиностроительного
производства

КУРСОВАЯ РАБОТА

по дисциплине «Конструкторско-технологическая информатика»
на тему: «Автоматизация расчета »

Выполнил студент 1-го курса группы БКТО-_01 _____
курс, группа, номер инициалы, фамилия
« » мая 202_ г.

Руководитель старший преподаватель каф. АОМП _____ С. Л. Новокшенов
должность, звание, подпись инициалы, фамилия
« » мая 202_ г.

Защищена _____ Оценка _____
Дата

Воронеж 202_

* Соответствие формы титульного листа принятому в настоящее время проверяйте на сайте ФГБОУ ВО ВГТУ

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
1. ПОСТАНОВКА ЗАДАЧИ.....	4
2. РЕШЕНИЕ ЗАДАЧИ	5
3. СОЗДАНИЕ АЛГОРИТМА ПРОГРАММЫ	7
4. РЕАЛИЗАЦИЯ АЛГОРИТМА НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ.....	11
ТЕМЫ КУРСОВОЙ РАБОТЫ.....	19
ТЕОРЕТИЧЕСКИЕ ВОПРОСЫ	23
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	24
ПРИЛОЖЕНИЕ. Титульный лист курсовой работы.....	26

КОНСТРУКТОРСКО-ТЕХНОЛОГИЧЕСКАЯ ИНФОРМАТИКА

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

к выполнению курсовой работы для студентов направления
15.03.05 «Конструкторско-технологическое обеспечение
машиностроительных производств» (профили «Технология
машиностроения», «Металлообрабатывающие станки
и комплексы», «Конструкторско-технологическое
обеспечение кузнечно-штамповочного производства»)
всех форм обучения

Составитель

Новокщенов Сергей Леонидович

Издается в авторской редакции

Подписано к изданию 25.11.2021.

Уч.-изд. л. 1,7.

ФГБОУ ВО «Воронежский государственный технический
университет»

394026, Воронеж, Московский просп., 14