Санкт-Петербургский государственный политехнический университет

Петра Великого

Институт прикладной математики и механики

**Кафедра «Теоретическая механика»**

**КУРСОВАЯ РАБОТА**

**Решение и оформление задачи по теоретической механике в TeХ, с использованием языка С**

по дисциплине «Языки программирования»

Выполнил

студент гр.23632/1 В.В.Филиппова

Руководитель

Ассистент А.Ю.Панченко

«31» мая 2018г.

Санкт-Петербург

2018

Содержание

Введение [3](#__RefHeading___Toc515469750)

Глава 1. Аналитическое решение задачи [4](#__RefHeading___Toc515469751)

1.1. Постановка задачи [4](#__RefHeading___Toc515469752)

1.2. Аналитическое решение задачи [5](#__RefHeading___Toc515469753)

Глава 2. Реализация задачи на языке программирования С [6](#__RefHeading___Toc515469754)

2.1. Принцип работы программы [6](#__RefHeading___Toc515469755)

2.2. Алгоритм работы программы [7](#__RefHeading___Toc515469756)

2.3. Использованные функции [8](#__RefHeading___Toc515469757)

2.4 Вывод программы [9](#__RefHeading___Toc515469758)

2.5. Выходные данные программы [1](#__RefHeading___Toc515469760)0

Заключение [1](#__RefHeading___Toc515469761)1

Список использованной литературы [1](#__RefHeading___Toc515469762)2

# **Введение**

Основной целью нашей работы является создание однотипных задач по теоретической механике, в данном случае мы генерируем задачу с вращением. Данная работа востребована, поскольку в результате мы получаем неограниченное число различных типовых задач с соответствующим решением.

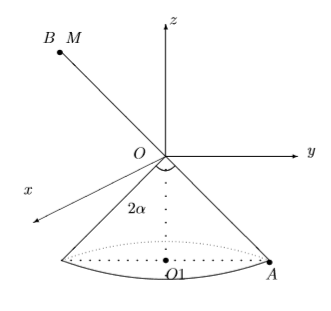
Вычисления в нашем проекте основываются на фундаментальных законах пространственной геометрии и теоретической механики.

Мы рассматриваем задачу со следующими характеристиками: конус неподвижно лежит на своём основании, диск закреплён на конусе посредством соединения центра масс диска и вершины самого конуса, диск вращается по конусу без проскальзывания.

В качестве объекта исследования был взят процесс программирования решения. В результате мы добились следующего: в качестве входных данных программа принимает файл формата .txt, в котором мы определяем значения известных по задаче параметров (относительную угловую скорость, относительное угловое ускорение, тангенсальную скорость точки М, тангенсальное ускорение точки М, радиус диска и угол; иллюстрация задачи приведена во входных данных), а также количество получаемых типовых вариантов (в данной работе генерируются 6 различных вариантов с соответствующими ответами). При завершении операций, описанных в нашей программе, на «выходе» мы получаем файл формата .tex, в котором содержатся исходные данные, и данные, которые необходимо найти (абсолютную угловую скорость, переносную угловую скорость, первую и вторую составляющую углового ускорения, вращательную скорость, первую и вторую составляющую вращательного ускорения, вращательное ускорение, ускорение точки, ускорение Кориолиса и ускорение точки М).

В качестве предмета исследования была взята задача по теоретической механики, описанная выше, для решения которой мы и написали программу.

# **Глава 1. Аналитическое решение задачи**

**1.1. Постановка задачи**

Условие задачи:

Диск АОВ обкатывает неподвижный конус. Также известны следующие параметры:

* **ОВ — радиус диска**
* **угол α**
* **Vr– относительная скорость т.М**
* **Wr– относительное ускорение т.М**
* **ω**e **- переносная угловая скорость диска**
* **εe – переносное угловое ускорение диска**

**Необходимо найти:**

* **собственную угловую скорость ωr**
* **абсолютную угловую скорость ωa**
* **абсолютное угловое ускорение εa**
* **скорость и ускорение точки B vB, wB**
* **абсолютную скорость и ускорение точки M vMа, wMа**

**1.2. Аналитическое решение задачи**

**1.Из** теоремы синусов находятся собственная угловая скорость и абсолютная угловая скорость:



Вектор абсолютной угловой скорости:

ωa = ωe + ωr

2. Вектор абсолютного углового ускорения определяется как производная по времени от вектора абсолютной угловой скорости:



3. Скорость точки B определяется как скорость этой точки во вращательном движении вокруг мгновенной оси по формуле

vB = v0 + ωa × OB = ωa × OB (т.к. v0 = 0),

где OB – радиус-вектор точки, проведённый из неподвижной точки О

Ускорение в точке B:

wB = w0 + εa×OB + ωa× ( ωa×OB ) = εa×OB + ωa × ( ωa×OB ) (т.к. w0 = 0)

4. Абсолютная скорость точки M складывается из скорости точки B и скорости движущейся по основанию конуса точки M:

vMа = vrM + vB

Абсолютное ускорение в точке M:

wMа = w0 + εa×OB + ωa×(ωa×OM) + wr + wcor, где

w0 + εa×OB + ωa×(ωa×OВ) = wB,

wcor = 2(ωa×vrM) – ускорение Кориолиса,

wr = wrn + wrτ — ускорение относительного движения

# **Глава 2. Реализация задачи на языке программирования С**

**2.1. Принцип работы программы**

**Для генерирования типовых вариантов задач, достаточно запустить программу. Она случайно выбирает входные параметры и решает по формулам, написанные в главе I.**

**2.2. Алгоритм работы программы**

**В качестве входных данных программа генерирует случайные значения, задаваемые в определенном диапазоне.**

**Далее программа открывает файл для чтения и template.tex. Этот файл представляет собой шаблон выходного файла с маркерами, с которыми впоследствии программа и работает.**

**in\_file = fopen("input\_file.txt", "r");**

textpl\_file = fopen("template.tex", "r");

**После данных действий программа заходит в цикл, где создаются и открываются шесть вариантов файлов типа .tex, содержащие условия рассматриваемой нами задачи с ответами и где выбираются случайным образом параметры, необходимые для решения задачи. В этом же цикле программа работает с маркерами, которые упоминались ранее.**

**Суть работы с маркерами состоит в следующем: внутри цикла программа начинает работать с шаблоном , считывая каждую стоку и копируя ее в созданный нами выходной файл типа .tex. Если в начале строки находится слово [MOD]**

**[MOD]\settext{Wb}{%.2f}**

**то эту строку копируем в выходной файл с пятого элемента, заменяя %.2f на параметр, который случайным образом генерируется. В результате данных действий получаем следующее выражение**

**\settext{Wb}{15},**

**которое запишется в выходной файл. В конце каждого цикла производится закрытие выходного файла.**

**2.3. Использованные функции**

**fopen("template.tex", "x") — функция, открывающая файл "template.tex" для чего-либо, в зависимости от параметра x.**

* **x = r – файл используется только для чтения, при отсутствии файла, создает его.**
* **x = w - файл используется только для записи, при отсутствии файла, создает его.**
* **x = r+ - файл используется как для чтения так и для записи, при отсутствии создает его.**
* **x = w+ - файл используется как для чтения так и для записи, при отсутствии создает его.**

**printf("Cannot open file: %x1\n", x2) — функция, печатающая в командную строку выражение "Cannot open file:" и при наличии второго выводимого слагаемого x2 необходимо вместо x1 ввести тип x2, при отсутствии второго выводимого слагаемого команда будет выглядеть следующим образом:**

**printf("Cannot open file\n",).**

**while(x) – цикл, который выполняется до тех пор, пока значение (x) -истинна; x – условие, которое позволяет войти в цикл.**

**rewind(textpl\_file) — функция, ставящая курсор в начало строки файла textpl\_file.**

**\*mod\_pos = strstr(&str3[0], "[MOD]") – функция, ищущая подстроку [MOD] в строке, которая находится в массиве str3 начиная с 0 позиции. А переменная mod\_pos сохраняет адрес первого символа искомой подстроки в строке str3.**

**fwrite(str, strlen(str), 1, out\_file) - функция, дописывающая в выходной файл строку, содержащуюся в массиве str.**

**fclose(textpl\_file) — функция, закрывающая файл textpl\_file.**

**2.4 Вывод программы**

**На выходе из программы мы получаем шесть различных файлов типа .tex с условием и соответствующими ответами, также присутствуют иллюстрации задач, которые были реализованы следующим образом.**

**Для начала необходимо задать координаты начальной точки, относительно которой будет построен рисунок:**

**\begin{picture}(20,20)(-290, 40),**

чтобы масштабировать рисунок, можно использовать команду

\setlength{\unitlength}{0.5mm}

далее начинается рисование с помощью следующих команд:

\put(-75,-20)

\put(0,0){\vector(0,1){70}}

\put(0,0){\line(-1,-1){55}}

\qbezier(-40,-53)(0,-75)(40,-53)

\multiput(0,0)(0,-7){10}%

{\circle\*{1}}

**Команда \put(x,y) позволяет рисовать точку с координатами (x,y) относительно выбранной ранее точки — центра рисунка.**

**Команда \put(x,y){\vector(x1,y1){n}} позволяет рисовать вектор, начало которого находится в точке (x,y) и концом в (x1,y1) длиной n.**

**Команда put(x,y){\line(x,y){n}} аналогична команде \put(x,y){\vector(x1,y1){n}}. Различие их состоит в том, что в первом случае рисуется линия, во втором — вектор.**

**Команда \qbezier(x1,y1)(x2,y2)(x3,y3) позволяет рисовать кривую по заданным точкам; (x1,y1) и (x2,y2) - крайние точки , (x3,y3) - промежуточная точка.**

**Команда \multiput(x1,y1)(x2,y3){N}% {\circle\*{N2}} позволяет рисовать пунктиром, где (x1,y1) — начало линии, (x2,y3) - расстояние между координатами по двум осям двух точек, N - количество точек, {\circle\*{N2}} — форма точек и их размер соответственно.**

**2.5. Выходные данные программы**



Рис. 1. Пример выходных данных

По окончанию работы, программа предоставляет 6 файлов типа .tex - варианты однотипных задач с ответами. При желании эти фалы можно скомпилировать, в результате чего получаем файлы типа .pdf (пример конвертированного файла показан на рис.1)

# **Заключение**

Для решения поставленной задачи было произведено распределение обязанностей: численное решение задачи было получено Кравченко И.С., программированием выходного файла формата .tex с условиями задачи и ответами занималась Филиппова В.В.; решением задачи движения диска по конусу занимались Кравченко И.С. и Филиппова В.В.  
В ходе данной работы были получены знания по работе с TeX и С, осуществлена работа с различными пакетами и командами. Для поставленной задачи был сделан рисунок, заданы начальные условия и вычислены значения собственной угловой скорости, абсолютной угловой скорости, абсолютного углового ускорения, скорости и ускорения точки B, абсолютной скорости и ускорения точки M.

**Список использованной литературы**

1. Онлайн компилятор файлов типа .tex: [https://ru.sharelatex.com](https://vk.com/away.php?to=https%3A%2F%2Fru.sharelatex.com&cc_key=)– (дата обращения 24.05.2018).
2. С. М. Львовский. Набор и верстка в системе LaTex – 2003: [https://www.mccme.ru/free-books/llang/newllang.pdf](https://vk.com/away.php?to=https%3A%2F%2Fwww.mccme.ru%2Ffree-books%2Fllang%2Fnewllang.pdf&cc_key=) – (дата обращения 24.05.2018).
3. Справочник по функциям по языкам программирования С/С++: [http://en.cppreference.com](https://vk.com/away.php?to=http%3A%2F%2Fen.cppreference.com&cc_key=) — (дата обращения 25.05.2018).
4. Брайан Керниган, Деннис Ритчи. Язык программирования С: второе издание, переработанное и дополненное. Изд-во Вильямс, 2017. — 288 с.