**Язык программирования**: C++

**Среда выполнения**: Microsoft Visual Studio

**Время выполнения**: 6 часов (время указано для ориентира, оно не является ограничением; главное – качество выполнения)

**Задача**:

Необходимо разработать консольное приложение, которое проводит тестирование работы двигателя с помощью 2-х тестов:

1. Тест рассчитывает и выводит на консоль время в *секундах*, которое пройдет от старта двигателя внутреннего сгорания до момента его перегрева.
2. Тест рассчитывает и выводит на консоль максимальную мощность двигателя в *кВт*, а также скорость вращения коленвала в *радиан/сек*, при котором эта максимальная мощность достигается.

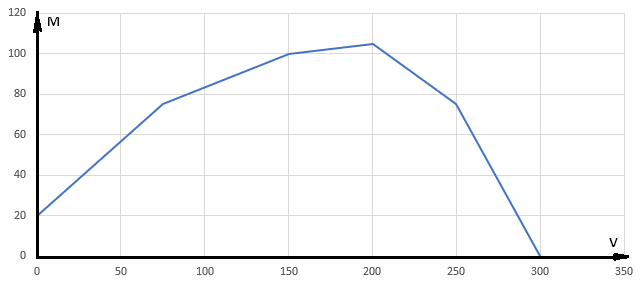
Рассчитывать результаты тестов точно, аналитическим путем не нужно, интересует получение этих значений методом симуляции. Разумеется, таким образом значения будут вычислены с определенной погрешностью. Плюсом будет возможность контролировать эту погрешность.

Приложение должно состоять из трех логических блоков:

1. **Симуляция двигателя внутреннего сгорания**

Требуется упрощенно симулировать изменение скорости вращения коленвала и температуры охлаждающей жидкости двигателя, работающего без нагрузки, с течением времени. Входные параметры:

* Момент инерции двигателя **I** (*кг∙м2*)
* Кусочно-линейная зависимость крутящего момента **M**, вырабатываемого двигателем, от скорости вращения коленвала **V** (крутящий момент в *Н∙м*, скорость вращения в *радиан/сек*):



* Температура перегрева **Tперегрева** (*C0*)
* Коэффициент зависимости скорости нагрева от крутящего момента **HM**
* Коэффициент зависимости скорости нагрева от скорости вращения коленвала **HV**
* Коэффициент зависимости скорости охлаждения от температуры двигателя и окружающей среды **C**

Так как двигатель работает без нагрузки, то весь вырабатываемый момент идет на раскрутку коленвала, и его ускорение вычисляется просто:

Специальной логики старта двигателя не требуется. Считаем, что при старте он просто начинает вырабатывать крутящий момент по заданному графику начиная с нулевой скорости вращения.

Скорость нагрева двигателя рассчитывать как **VH = M × HM + V2 × HV** (*С0/сек*)

Скорость охлаждения двигателя рассчитывать как **VC = C × (Tсреды - Тдвигателя)** (*С0/сек*)

Температура двигателя до момента старта должна равняться температуре окружающей среды. Нагрев и охлаждение, рассчитанные по формулам выше, действуют на двигатель постоянно, одновременно и независимо друг от друга.

Мощность двигателя внутреннего сгорания рассчитывать как **N = M × V / 1000** (*кВт*)

1. **Логика тестирования двигателя**

Требуется реализовать 2 «тестовых стенда», исследующие поданный на вход двигатель:

1. Тестовый стенд нагрева должен включать двигатель, следить за его температурой, и в момент перегрева прекращать тест и выдавать время, прошедшее с момента старта до перегрева.
2. Тестовый стенд максимальной мощности должен включать двигатель и снимать с него показания до того момента пока двигатель не перестанет раскручиваться. На этом тест должен завершаться и выдавать, какая максимальная мощность двигателя была достигнута, и при какой скорости коленвала.

Расчет симуляции двигателя не должен производиться в реальном времени. Необходимо использовать модельное время, чтобы ожидание результатов работы программы не было продолжительным.

1. **Консольный ввод-вывод, задание исходных данных и запуск теста**

Эта часть приложения содержит точку входа, и должна обеспечивать весь ввод/вывод на консоль, а также задание всех исходных данных и запуск теста двигателя. Все исходные данные, кроме температуры окружающей среды, нужно задать в коде или в конфигурационном файле:

I = 10

M = { 20, 75, 100, 105, 75, 0 } при V = { 0, 75, 150, 200, 250, 300 } соответственно

Tперегрева = 110

HM = 0.01

HV = 0.0001

C = 0.1

Температура окружающей среды вводится пользователем с клавиатуры в градусах Цельсия, после запуска приложения.

**Критерии оценки**:

Выполненное задание будет оцениваться по следующим критериям:

1. Корректность работы. Приложение должно выдавать корректный результат на различных входных данных, не должно быть вылетов и зависаний.
2. Архитектура. Простота расширения функционала: добавление новых типов двигателей, в том числе не внутреннего сгорания, новых типов тестов. Простота изменения входных данных тестов.
3. Оформление. Читаемость кода, организация файлов и папок приложения.
4. Знание языка программирования C++

В качестве результата необходимо выслать архив, содержащий проект MS Visual Studio и файлы исходного кода (либо ссылку на скачивание архива).