**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УІВЕРСИТЕТ ІМЕНІ І. І. МЕЧНИКОВА ІНСТИТУТ МАТЕМАТИКИ, ЕКОНОМІКИ І МЕХАНІКИ**

Лабораторная работа №2:

" Разработка модельного интеллектуального пакета прикладных программ (ИППП)"

Выполнил: студент 3 курса ПМ

Никифоров Алексей

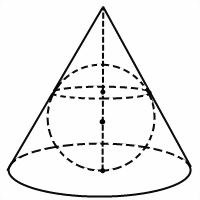
Одесса, 2016

**Цель работы**

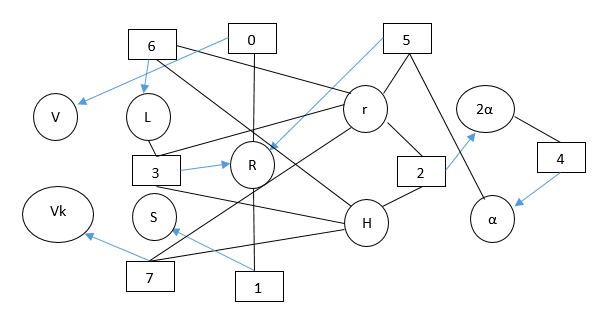
Проектирование и разработка некоторых элементов модельного ИППП для автоматического построения цепочки программных модулей для решения поставленной задачи в предметной области (ПО) пакета.

Вариант: 13 Шар вписанный в конус

|  |  |
| --- | --- |
| ***Идентификатор атрибута*** | ***Название атрибута*** |
| r | Радиус конуса |
| R | Радиус Шара |
| H | Высота конуса |
| t\_2alpha | Тангенс угола при основании |
| L | Боковая сторона |
| V | Объем шара |
| S | Площадь поверхности шара |
| Vk | Объем Конуса |



|  |  |  |
| --- | --- | --- |
| ***#*** | ***Отношение*** | ***Параметры*** |
| 0 | V = { R > 0 | } | R |
| 1 | S = { R > 0 | | R |
| 2 | t\_2alpha = { H, r > 0| } | H, r |
| 3 | R = {H, r, L > 0| | H, r, L |
| 4 | t\_alpha = { } | t\_2alpha |
| 5 | R = {r > 0 | | r, t\_2alpha |
| 6 | L = | r, H |
| 7 | Vk = | r, H |



Все возможные решения представлены в графе.

**Решение:**

Основные структуры данных:

TrpObject – объект в памяти, содержит идентификатор, полное имя и значение, которые вычисляется единожды.

public string Ident { get; private set; }

public string Name { get; private set; }

private double \_value;

ModulInterface – описание интерфейса модуля, содержит идентификатор, полное имя, возвращаемое значение и набор параметров.

public string ModuleIdent { get;}

public string ModuleName { get;}

public string ModuleOutput { get; }

public List<string> ModulParams { get; }

Rp – рабочая память, заполняется из файла (M.txt) объектами типа TrpObject. Объектам в памяти ставятся в однозначное соответствие порядковый номер.

Mi – функциональное наполнение пакета, заполняется из файла объектами типа ModulInterface. Так же тело функции берем из файла (B.txt). Тело парсим при помощи библиотеки (ELW.Library.Math автор Д. Заславский).

Is – логическое ядро модуля содержит основную логику, которую мы обсудим в след пункте.

Ippp – объединяет в себе все элементы системы. По сути предоставляет в себе API для работы с IS и дает возможность добавлять модули, выгружать объекты из файлов и парсить их при помощи внешней библиотеки. Таким образом имеем не зависящий от способа представления функции класс Is и его конкретную обертку Ippp.

Данные подаются в файл data.txt в формате пар значений <имя:значение> и пары которая ставит задачу в виде <имя:?>

Основные методы и идея алгоритма:

После того как данные выгружены в память остается составить алгоритм, которые позволяет говорить о возможности решения с последующим решением. Идея алгоритма реализованного внутри ядра Is проста – найти “nodes(звенья)” ведущие к решению и применить этот же алгоритм к подзадаче меньшей размерности. При это нужно помечать nodes для того чтобы не вычислять повторно. После такого заполнения мы придём либо к случаю тупикового решения или к решению. В случае не валидности входных данных очень сложно оценить какой параметр привел к не правильному решению, поэтому в таком случае будем использовать проверку на уровне Ippp модуля на несоответствие. Так же модуль Is отвечает за десериализацию объектов и объединяет в себе функциональное наполнение пакета и рабочую память.

Пример работы программы:

#1

1: r : r

2: R : R

3: H : H

4: t2a : t2a

5: L : L

6: V : V

7: S : S

8: Vk : Vk

9: ta: ta

#2

V = F1(R) : Calc V

S = F2(R) : Calc S

t2a = F3(H, r) : Calc t2a

R = F4(H, L, r) : Calc R

ta = F5(t2a) : Calc t2a

R = F6(r, ta) : Calc R

L = F7(r,H) : Calc L

Vk = F8(r, H) : Calc Vk

F1 V=4\*3.14\*R^3/3

F2 S=4\*3.14\*R^2

F3 t2a=H/r

F4 R=(H+r)/(L+r)

F5 ta=t2a/(1+(1+t2a^2)^0.5)

F6 R=r\*ta

F7 L=(r^2+H^2)^(1/2)

F8 Vk=(3.14\*r^2\*H)/3

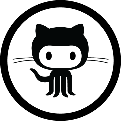
r=1

t2a=0.5

V-? => V = 0.78677; depth = 2;

Таким образом мы можем убедится, что наш поиск рекурсивный поиск работает не зависимо от требуемой глубины.

You can find code here:

[](https://github.com/NikiforovAll/AICourse)

https://github.com/NikiforovAll/AICourse