

Лабораторная работа №3

ЯЗЫК PLUS И ЭКСПЕРИМЕНТЫ В СРЕДЕ GPSS WORLD

Цель работы – изучение встроенного алгоритмического языка PLUS и основ работы с экспериментами в среде GPSS World

1. Теоретическая часть

Слабость алгоритмических средств долгое время была недостатком GPSS. Однако, начиная с GPSS World, язык моделирования GPSS был дополнен простым встроенным алгоритмическим языком PLUS (Programming Language Under Simulation). В виде подпрограмм языка PLUS могут быть реализованы программные конструкции, отражающие особенности поведения моделируемых объектов. Еще одной, не менее важной, областью применения языка PLUS является разработка экспериментов – специальных пользовательских процедур, реализующих специфическую логику управления моделированием.

1.1. Основы языка PLUS

Язык PLUS содержит основные операторы, характерные для процедурных алгоритмических языков высокого уровня. Так же язык PLUS включает средства для чтения/записи данных в файлы, для обращения к внешним программам, хранящимся в виде файлов EXE или DLL, для управления моделированием.

1.1.1. Алфавит языка PLUS

В языке PLUS допускается использовать алфавитно-цифровые символы в именах и специальные символы в качестве знаков операций и пунктуации.

К алфавитно-цифровым символам относятся буквы латинского алфавита в нижнем “a”-“z” или верхнем “A”-“Z” регистрах, цифры “0”-“9”, а так же символ подчеркивания “_”.

К специальным символам относятся символы “#”, “*”, “&”, “+”, “-”, “/”, “\”, “.”, “:”, “^”, “(”, “)”, “ ”, “[”, “]”.

1.1.2. Имена и выражения языка PLUS. Типы данных

Имена в языке PLUS используются для обозначения объектов, переменных и процедур. Имя должно начинаться с буквы и содер-

жать до 255 алфавитно-цифровых символов. Символы верхнего и нижнего регистра не различаются.

В тексте программы могут присутствовать комментарии, заключенные в составные скобки /* */.

Выражения в языке PLUS могут содержать имена переменных и объектов, операции и вызовы процедур. Основные операции языка PLUS описаны в табл. 1.

Табл. 1. Основные операции языка PLUS

№ п/п	Операция	Результат	Тип
Арифметические операции			
1	-	Изменение знака на противоположный	Унарная
2	^	Возведение в степень	Бинарная
3	#	Умножение	Бинарная
4	/	Деление	Бинарная
5	\	Целочисленное деление	Бинарная
6	@	Остаток от деления	Бинарная
7	+	Сложение	Бинарная
8	-	Вычитание	Бинарная
Логические операции			
9	NOT или ~	Инверсия	Унарная
10	AND	Логическое И	Бинарная
11	OR	Логическое ИЛИ	Бинарная
12	>	Больше	Бинарная
13	<	Меньше	Бинарная
14	=	Равно	Бинарная
15	/=	Не равно	Бинарная
16	<=	Меньше или равно	Бинарная
17	>=	Больше или равно	Бинарная

Переменные пользователя и параметры объектов могут иметь значения различных типов данных. В GPSS World используются три основных типа данных:

Целочисленный (Integer). 32-х битовое двоичное число. В процессе выполнения математических операций может быть автоматически преобразовано в вещественное.

Вещественный (Real). Число с плавающей точкой двойной точности.

Строковый (String). Последовательность символов ASCII. Максимальная длина строки задается в настройках системы.

1.1.3. Операторы языка PLUS

Набор операторов языка PLUS примерно соответствует основным операторам процедурного языка высокого уровня. Операторы языка PLUS описаны ниже.

1) Оператор присваивания

Имеет формат:

`<Переменная или элемент матрицы> = <выражение>;`

Например:

`Count = 1;`

Переменная или элемент матрицы получает значение, определяемое выражением.

2) Составной оператор

Имеет формат:

`BEGIN <Операторы> END;`

Например:

```
BEGIN
    Count = 1;
    NewValue = OldValue;
END;
```

Создает блок PLUS-операторов.

3) Оператор EXPERIMENT

Имеет формат:

`EXPERIMENT <Имя> (<Список параметров>) <Оператор>;`

Создает PLUS-эксперимент – особый вид процедуры, предназначенный для автоматизации исследования модели. Подробнее эксперименты рассматриваются в п. 1.3.

4) Оператор GOTO

Имеет формат:

`GOTO <Метка>;`

Например:

`GOTO BeginLabel;`

Выполняет переход к оператору, помеченному указанной меткой.

5) Условный оператор

Имеет формат:

```
IF (<Выражение>) THEN <Оператор1> ELSE <Оператор2>
      IF (<Выражение>) THEN <Оператор1>
```

Например:

```
IF (Count<=5) THEN BEGIN
    Count = Count+1;
    B = "TRUE";
END;
```

Проверяет результат выражения, и если он равен "TRUE", то выполняется первый оператор. Если результат выражения равен "FALSE", то выполняется второй оператор. Условный имеет полную и сокращенную (без ветви ELSE) форму.

6) Помеченный оператор

Имеет формат:

```
<Метка> : <Оператор>
```

Например:

```
BeginLabel : Count = 0;
```

Используется для перехода в операторе GOTO.

7) Оператор PROCEDURE

Имеет формат:

```
PROCEDURE <Имя> (<Список параметров>) <Оператор>;
```

Например:

```
PROCEDURE PlusOne(Number) BEGIN
    RETURN Number + 1;
END;
```

Определяет пользовательскую процедуру. Процедура может стоять в любом месте модели, предшествующем месту обращения к процедуре.

8) Вызов процедуры

Имеет формат:

```
<Имя процедуры> (<Список выражений>);
```

Например:

```
Exponential(1, 0, 1/lambda);
```

Используется для вызова стандартной или пользовательской процедуры с указанными параметрами.

9) Оператор RETURN

Имеет формат:

```
RETURN <Выражение>;
```

Например:

```
RETURN Number + 1;
```

Прерывает выполнение процедуры и задает ее результирующее значение равным значению выражения.

10) Оператор TEMPORARY

Имеет формат:

```
TEMPORARY <Список переменных>;
```

```
TEMPORARY MATRIX <Имя>[<Список размерностей>]
```

Например:

```
TEMPORARY I, J;
TEMPORARY MATRIX M[2, 3];
```

Объявляет локальные переменные или матрицу PLUS-процедуры. При объявлении матрицы число индексов должно быть в пределах от 1 до 6.

11) Оператор WHILE

Имеет формат:

```
WHILE (<Условие>) DO <Оператор>
```

Например:

```
WHILE (I>0) DO BEGIN
    B = B*2;
    I = I-1;
END;
```

Реализует цикл с предусловием. Если условие истинно, то выполняется очередная итерация цикла.

1.1.4. Встроенные процедуры языка PLUS

GPSS World содержит большое количество встроенных процедур, часть из которых перечислены в табл. 2.

Табл. 2. Стандартные процедуры языка PLUS

№ п/п	Формат	Описание
Математические процедуры		
1	ABS(<Выражение>)	Абсолютная величина числа
2	ATN(<Выражение>)	Арктангенс (в радианах)
3	COS(<Выражение>)	Косинус
4	EXP(<Выражение>)	Число е, возведенное в степень аргумента
5	INT(<Выражение>)	Целая часть числа
6	LOG(<Выражение>)	Натуральный логарифм
7	SIN(<Выражение>)	Синус
8	SQR(<Выражение>)	Квадратный корень
9	TAN(<Выражение>)	Тангенс
Строковые процедуры		
10	Catenate(<Строка1>, <Строка2>)	Производит объединение двух строк
11	Find(<Строка1>, <Строка2>)	Вычисляет смещение одной строки, содержащейся в другой строке
12	Length(<Строка>)	Возвращает количество символов в строке
13	LowerCase(<Строка>)	Переводит все символы строки в нижний регистр
14	Place(<Строка1>, <Строка2>, <Смещение>)	Вставляет одну строку в другую
15	Polycatenate(<Список строк>)	Производит объединение нескольких строк
16	String(<Данные>)	Преобразует данные в их строковый эквивалент
17	Substring(<Строка>, <Смещение>, <Число символов>)	Возвращает подстроку исходной строки
18	UpperCase(<Строка>)	Переводит все символы строки в верхний регистр
19	Value(<Строка>)	Возвращает числовой эквивалент строки
Распределения вероятностей		
20	UNIFORM(<поток>, <минимум>, <максимум>)	Равномерно распределенная величина

21	EXPONENTIAL(<поток>, <смещение>, <параметр распределения>)	Экспоненциально распределенная величина
22	DUNIFORM(<Поток>, <минимум>, <максимум>)	Дискретное равновероятное распределение в заданном диапазоне натуральных чисел

1.2. Пример использования языка PLUS в модели на GPSS

В качестве примера рассмотрим модель кассира в супермаркете. Будем считать, что каждый покупатель может взять от 1 до 10 единиц разных товаров. Время учета единицы какого либо товара кассиром распределено по экспоненциальному закону с интенсивностью λ . Время оплаты покупки распределено по экспоненциальному закону с интенсивностью μ . Интервал между поступлениями покупателей на кассу распределен экспоненциально с интенсивностью ν .

Соответствующая модель на языке GPSS имеет вид:

```

Greek_lambda EQU      20
Greek_mu     EQU      10
Greek_nu     EQU      1
              GENERATE (Exponential(1, 0, 1/Greek_nu))
              QUEUE    Ochered
              SEIZE     Kassir
              DEPART    Ochered
              ADVANCE   (KassirDelay())
              RELEASE   Kassir
              TERMINATE 1
PROCEDURE KassirDelay() BEGIN
    TEMPORARY FullDelay;
    FullDelay = 0;
    TEMPORARY Idx, ByingCount;
    Idx = 1;
    ByingCount = DUNIFORM( 1, 1, 10 );
    WHILE (Idx<=ByingCount) DO BEGIN
        FullDelay = FullDelay + Exponential(1, 0, 1/Greek_lambda);
        Idx = Idx + 1;
    END;
    FullDelay = FullDelay + Exponential(1, 0, 1/Greek_mu);
    RETURN FullDelay;
END;
```

1.3. Эксперименты в среде GPSS World

Когда модель исследуемой системы разработана и протестирована, наступает фаза проведения экспериментов над разработанной моделью для выявления тех или иных свойств модели. В языке PLUS

имеется специальный вид процедуры, предназначенный для автоматизации проведения экспериментов над имитационной моделью, который так и называется – эксперимент. В GPSS World существует два типа экспериментов – пользовательские и встроенные. Пользовательские эксперименты пишутся пользователем на языке PLUS вручную. Программный код для встроенных экспериментов генерируется системой автоматически по значениям параметров, заданным пользователем в соответствующих диалоговых окнах GPSS World.

В GPSS World существует два типа пользовательских экспериментов – отсеивающий эксперимент и оптимизирующий эксперимент. Отсеивающий эксперимент (Screening Experiment) предназначен для выявления степеней влияния на характеристики модели ее входных параметров. Оптимизирующий эксперимент (Optimizing Experiment) строит поверхность отклика модели и находит оптимум заданного критерия и соответствующие оптимальные значения модельных параметров.

Эксперимент представляет собой особый вид PLUS-процедуры. Вместо зарезервированного слова PROCEDURE при объявлении эксперимента используется зарезервированное слово EXPERIMENT. Из эксперимента и из процедур, вызываемых из эксперимента (любого уровня вложенности), может вызываться служебная процедура DoCommand. Данная процедура выполняет команду GPSS, которая задается единственным строковым параметром процедуры. Вызывая данную процедуру с различными параметрами, можно осуществить довольно сложные манипуляции с моделью. Например, выполнить несколько имитационных прогонов с различными значениями параметров модели и сравнить значения полученных характеристик и т.п.

Для того чтобы запустить эксперимент, нужно откомпилировать модель, а затем выбрать пункт меню “Command\CONDUCT”. В появившемся окне “Conduct Experiment Command” нужно дописать имя эксперимента и список значений параметров, и нажать кнопку “OK”.

Для того чтобы во время выполнения эксперимента выдать промежуточные данные в лист протокола, можно вызвать команду GPSS SHOW, например:

```
DoCommand( "SHOW ""User experiment is running now... "" " )
```


1.4. Пример пользовательского эксперимента

Рассмотрим следующий пример эксперимента. Пусть требуется исследовать несколько вариантов реализации СМО с числом каналов K от 1 до 8 и средним временем обслуживания одного канала τ от 1 до 8, причем $K/\tau = \text{const}$ (общая производительность системы остается постоянной). Требуется выбрать вариант с наименьшим итоговым коэффициентом загрузки СМО. Интервал поступления заявок и время обслуживания распределены равномерно в диапазоне $\pm 30\%$ от среднего значения. Соответствующая программа на GPSS имеет вид:

```
t_vh      EQU      1
tau       EQU      8
Smo       STORAGE  1
GENERATE   t_vh, (t_vh#0.3)
ENTER     Smo, 1
ADVANCE    tau, (tau#0.3)
LEAVE     Smo, 1
TERMINATE  1

EXPERIMENT TestSMO() BEGIN
  TEMPORARY Idx; /* Параметр цикла */
  TEMPORARY Str; /* Строка для формирования команд */
  TEMPORARY BestIdx; /* Лучший вариант системы */
  TEMPORARY BestUtilization; /* Лучший коэффициент использования */
  Idx = 1; /* Присвоение начальных значений */
  BestIdx = 0;
  BestUtilization = 1000;
  WHILE (Idx<=8) DO BEGIN /* Цикл по Idx = 1 .. 8 */
    /* Начало моделирования очередного варианта СМО */
    /* Выдача в протокол значений параметров модели */
    DoCommand("SHOW """);
    Str = Polycatenate("SHOW ""Model is runing with Idx=",
                      String(Idx),
                      "");
    DoCommand(Str);
    Str = Polycatenate("SHOW "" tau = K = ",
                      String(Idx),
                      "");
    DoCommand(Str);
    /* Сброс статистики */
    DoCommand("CLEAR OFF");
    /* Корректировка параметров модели */
    Str = Catenate("tau EQU ", String(Idx));
    DoCommand(Str);
    Str = Catenate("Smo STORAGE ", String(Idx));
    DoCommand(Str);
    /* Запуск моделирования */
    DoCommand("START 1000,NP");
    /* Выдача результата - коэффициента использования */
    Str = Polycatenate("SHOW "" SMO UTILIZATION = ",
                      String(SR$Smo),
                      "");
    DoCommand(Str);
    /* Определение наилучшего варианта системы */
```

```

IF (BestUtilization>SR$Smo) THEN BEGIN
    BestIdx = Idx;
    BestUtilization = SR$Smo;
END;
/* Конец имитационного прогона варианта СМО */
Idx = Idx+1;
END;
/* Выдача наилучшего варианта системы */
DoCommand("SHOW """);
Str = Polycatenate("SHOW "" BestIdx = ",
                  String(BestIdx),
                  "");
DoCommand(Str);
END;

```

Комментарии приведены в тексте программы. Необходимо иметь в виду, что комментарии на русском языке приведены исключительно в учебных целях, и в GPSS World они **запрещены**. Программа с русскими комментариями компилироваться не будет!

2. Порядок выполнения работы

1. Изучите теоретический материал.
2. Разработайте GPSS-эксперимент в соответствии с заданием для вашего варианта.
3. Выполните моделирование и определите необходимые результаты.
4. Сформулируйте выводы по работе и ответьте на контрольные вопросы.

3. Содержание отчета

1. Название работы.
2. Цель работы.
3. Задание на лабораторную работу.
4. Описание основ языка PLUS.
5. Программа на языке GPSS.
6. Результаты моделирования.
7. Выводы по работе.

4. Задание на лабораторную работу

Требуется исследовать работу многоканальной СМО. Число каналов СМО равно K . Интервал поступления заявок на вход СМО экспоненциально распределен с интенсивностью λ , время обслуживания фиксированное – τ . Разработайте GPSS-эксперимент в соот-

ветствии со следующими данными (за один прогон необходимо выполнить моделирование обслуживания 500 транзактов):

Номер варианта	K	λ	τ	Задача исследования
1	3	-	5	Требуется найти такое минимальное значение λ , для которого коэффициент загрузки СМО превышает 0,75. Шаг изменения λ взять равным 0,05.
2	-	10	1	Требуется найти такое минимальное число каналов K , при котором коэффициент загрузки СМО не превышает 0,75.
3	3	8	-	Требуется найти такое максимальное значение τ , для которого коэффициент загрузки не превышает 0,65. Шаг изменения τ взять равным 0,25.
4	3	-	2	Требуется найти такое минимальное значение λ , для которого коэффициент загрузки СМО превышает 0,8. Шаг изменения λ взять равным 0,025.
5	-	15	0,5	Требуется найти такое минимальное число каналов K , при котором коэффициент загрузки СМО не превышает 0,8.
6	2	10	-	Требуется найти такое максимальное значение τ , для которого коэффициент загрузки не превышает 0,8. Шаг изменения τ взять равным 0,3.
7	2	-	1	Требуется найти такое минимальное значение λ , для которого коэффициент загрузки СМО превышает 0,95. Шаг изменения λ взять равным 0,05.
8	-	10	2	Требуется найти такое минимальное число каналов K , при котором коэффициент загрузки СМО не превышает 0,85.

9	3	12	-	Требуется найти такое максимальное значение τ , для которого коэффициент загрузки не превышает 0,75. Шаг изменения τ взять равным 0,25.
10	2	-	7	Требуется найти такое минимальное значение λ , для которого коэффициент загрузки СМО превышает 0,9. Шаг изменения λ взять равным 0,025.

5. Контрольные вопросы

1. Для чего предназначен язык PLUS?
 2. Алфавит языка PLUS. Имена и выражения в языке PLUS.
- Типы данных.
3. Основные операции языка PLUS.
 4. Операторы языка PLUS.
 5. Типы экспериментов в GPSS World.
 6. Средства языка PLUS для реализации экспериментов.

Библиографический список

1. Томашевский В., Жданова Е. Имитационное моделирование в среде GPSS. – М.: Бестселлер, 2003. – 416 с.