

Л и т е р а т у р а

1. М а л и н и н Н.Н. Прикладная теория пластичности и ползучести. М., "Машиностроение", 1968.
2. Р а д ц и г Ю.А. Статически неопределимые фермы наименьшего веса. М., Изд -во Казанского университета, 1969.
3. И л ь ю ш и н А.А. Пластичность. ОГИЗ, ГИТТЛ, 1948.

УДК 681.3.065

А.И. Д а н и л и н

СОЗДАНИЕ ВХОДНОГО ЯЗЫКА ПОЛЬЗОВАТЕЛЯ В АВТОМАТИЗИРОВАННЫХ СИСТЕМАХ ПРОЕКТИРОВАНИЯ

При создании автоматизированных систем проектирования возникает проблема управления расчетными модулями с сохранением таких важных свойств системы как открытость и гибкость использования [1]. Традиционные управляющие программы реализуют жесткую последовательность вызова модулей, при добавлении новых модулей необходима корректировка управляющей программы. Поэтому большинство разработчиков пришли к идее генерации управляющих программ и в этой связи к необходимости входного языка пользователя [2], [3]. Язык в виде меню обеспечивает гибкость использования только для отдельных модулей. При добавлении новых модулей естественно необходимо введение новых элементов в язык и, следовательно, изменение транслятора.

При программировании на ПЛ/1 имеется возможность записи управляющей программы, не накладывающей никаких ограничений на последовательность и количество вызываемых модулей. Такие управляющие программы практически реализуют входной язык пользователя, но при этом возникает необходимость написания транслятора с этого языка к, соответственно, исправлений при добавлении новых моду-

лей. Для этой цели имеет средства создания программных фаз с перекрытием. Для этой цели служит оператор *CALL OVERLAY* ('имя фазы');

который из библиотеки абсолютных модулей загружает в оперативную память фазу с указанным именем по адресу, определенному во время редактирования. Далее этой фазе управления должно быть передано с помощью оператора *CALL* имя точки входа (список параметров). Имя фазы в операторе *CALL OVERLAY* должно быть уникальным, тогда как имя точки входа из оператора *CALL* не обладает таким свойством. Разные фазы могут иметь одинаковые имена точек входов, важно только, чтобы всем одинаковым именам входов при редактировании присвоились одинаковые адреса. Разбор машинной программы, сгенерированной транслятором ПЛ/И ДЭС ЕС ЭВМ показал, что в операторе *CALL OVERLAY* вместо имени фазы в апострофах можно указывать имена проблемных переменных без апострофов. Этим проблемным переменным можно обычным образом присвоить значения или ввести с перфокарт или другого внешнего носителя.

Для примера рассмотрим текст управляющей программы, в которой имя программы вводится с перфокарт:

```
UP: PROCEDURE OPTIONS (MAIN);  
  DECLARE PROGR CHARACTER (6);  
  M1: GET EDIT (PROGR) (COLUMN (1), A (6));  
  IF PROGR = "STOP" THEN STOP;  
  CALL OVERLAY (PROGR); CALL MODUL;  
  GOTO M1;  
  END UP.
```

Здесь имя вызываемой программы вводится в поле памяти, ответственной переменной *PROGR*. Имя содержит шесть алфавитно-цифровых символов и программа с этим именем должна находиться в библиотеке абсолютных модулей. Оператор *CALL OVERLAY (PROGR)* загружает указанную программу в оперативную память. Далее ей передается управление и после отработки происходит переход на ввод имени следующей программы, которую нужно выполнить. Все модули, вызываемые приведенной управляющей программой должны иметь имя точки входа: *MODUL* и в данном случае не должны содержать списка параметров.

Попутно заметим, обмен информацией между модулями лучше про-

водить через внешнюю память, нежели через параметры [4].

Пользователь должен на перфокартах набить имена программ, которые должны быть вызваны и выполнены. Таким образом, мы получаем язык директив. Последовательность вызова программ произвольна, ее определяет пользователь.

Предлагаемая техника управляющих программ опробована в ДЭС ЕС версий 1.3 и 2.1 при написании комплекса программ, реализующего метод конечных элементов. Удалось создать открытую систему в смысле использования номенклатуры конечных элементов. Для добавления новых типов элементов необходимо лишь дописать программу деформирования матрицы жесткости и выдачи результатов. Остальные программы остаются неизменными. Номенклатура конечных элементов выросла с одного до пяти типов и количество программ возросло с 16 до 32. При этом необходимость в корректировке управляющих и ранее написанных программ не возникла ни разу.

Л и т е р а т у р а

1. М а з у р и к В.П. Управление расчетными модулями в многоязыковых системах автоматизации проектирования. — Труды МФТИ. Серия "Аэрофизика и прикладная математика". Долгопрудный, 1977, с. 155-161.
2. М а з у р и к В.П., М е д в е д е в А.Е., С у ш - к о в Б.Г. Информационно-расчетная система для автоматизированного проектирования. Труды XX научной конференции МФТИ - 1974. Серия "Аэрофизика и прикладная математика", часть II, Долгопрудный, 1975, с. 3-33.
3. С т о л я р о в Л.Н. Принципы организации пакетов прикладных программ. Материалы Всероссийской школы 1975 года по автоматизации проектирования. Под ред. Н.Н. Моисеева. МФТИ, 1976, с. 127-141.
4. Итоги науки и техники. "Авиастроение", т.3. Машинное проектирование летательных аппаратов. ВИНТИ, М., 1976, с. 215.