

Ц840-В

А-482

СООБЩЕНИЯ  
ОБЪЕДИНЕННОГО  
ИНСТИТУТА  
ЯДЕРНЫХ  
ИССЛЕДОВАНИЙ

Дубна

3440/1-71



11-5963

ЛАБОРАТОРИЯ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ  
И АВТОМАТИЗАЦИИ

Л.И. Лепилова, А.Ф. Лукьянцев

ПРОГРАММА СТАТИСТИЧЕСКОГО АНАЛИЗА  
ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ

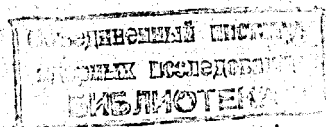
НА ЭВМ БЭСМ-6

1971

11-5963

Л. И. Лепилова, А. Ф. Лукьянцев

ПРОГРАММА СТАТИСТИЧЕСКОГО АНАЛИЗА  
ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ  
НА ЭВМ БЭСМ-6



Программа статистического анализа результатов эксперимента *SUMX* /1,3/ является одной из программ системы обработки экспериментальных данных<sup>4/</sup>. Она предполагает, что анализируемые данные находятся на магнитной ленте *DST* (*Data Summary Tape* ) и считывает их оттуда событие за событием для построения гистограмм, идеограмм, двумерных диаграмм рассеяния, для нахождения средних значений указанных величин и их ошибок, для суммирования, умножения, вычитания и деления гистограмм и т.д. Задавая различные критерии (тесты) программе, можно выбирать только определенные классы событий из множества событий на магнитных лентах *DST*. Имеется возможность нестандартным образом досчитывать некоторые физические величины, которые по каким-либо причинам не были записаны ранее на *DST*. Данная программа может использоваться не только для анализа информации, полученной с пузырьковых камер, но и для анализа любой статистической информации.

Программа имеет блочную структуру, и каждый блок выполняет определенные функции. Перечислим основные блоки, из которых состоит программа *SUMX*.

1) Блок *SELECT* предназначен для вычисления значений задаваемых тестов для каждого события.

2) Блок *CHARM* позволяет вычислять некоторые физические величины, которые не содержатся в явном виде на *DST*. Однако такие величины можно вычислить, используя находящуюся на *DST* информацию. Вычисляемые величины помещаются на специальное место в памяти ЭВМ за массивом, считанном с *DST*, затем, наряду с остальными, подвергаются проверке по всевозможным тестам. Подпрограммы, выполняющие такие вычисления, называются *CHARM1*, ..., *CHARM9* и состав-

ляются пользователем. Поскольку все эти подпрограммы являются частью *SUMX*, следует иметь в виду, что вычисление дополнительных величин может привести к большим затратам машинного времени. Кроме того, подпрограммы *CHARM* могут занять значительную часть оперативной памяти ЭВМ.

3) *BLOC 6* служит для накопления, печати и выдачи в виде графиков гистограмм и идеограмм для различных величин с соответствующими весами.

4) *BLOC 7* накапливает и выдает на печать двумерные гистограммы для величин, находящихся на *DST*. Дополнительно накапливаются и выдаются на печать одномерные гистограммы проекций на координатные оси. Имеется возможность строить контур, а на одномерных гистограммах проекций на оси рисовать фазовую кривую.

5) *BLOC 4* суммирует, умножает и делит гистограммы, накопленные в *BLOC 6*.

6) *BLOC 10* находит средние значения и стандартные отклонения указанных на *DST* величин по тем событиям, для которых выполняются заданные тесты.

7) *BLOC 14* и *BLOC 15* служат для составления и вывода на печать списков событий, удовлетворяющих заданным тестам, списков хороших или отбрасываемых событий и другой полезной информации. Использование этих блоков пользователем, который нечетко представляет алгоритм их работы, может привести к значительному увеличению машинного времени.

Порядок прохождения события через различные блоки весьма существенен, поскольку результаты работы одного блока обычно используются другим, как, например, тест-вектор\*, составленный блоком

*SELECT* для события, используется затем всеми последующими блоками при анализе значений тестов.

Входной информацией для *SUMX* являются:

1) магнитные ленты *DST* с результатами эксперимента, получаемые в результате работы программы *SLICE*<sup>1,2</sup> или любой, аналогичной *SLICE*, программы; 2) подпрограммы *CHARM*; 3) управляющие карты (УПК), карты запроса и информационные карты (ИПК).

\* Под тест-вектором понимаем результат проверки всех заданных тестов, составленных по определенным правилам.

УПК, в первом столбце которых пробивается ж, определяют режим работы *SUMX*.

Карты запроса являются подмножеством УПК и определяют блоки программы, используемые в дальнейшем.

На ИПК задаются детальные операции, которые должны выполнить блоки. Подробное описание УПК и ИПК дается в работе<sup>/1/</sup>.

Далее мы остановимся на общей схеме работы *SUMX*, на мнемоническом задании номеров тестов и адресов величин на *DST*, определении тестов и накоплении гистограмм.

### § I. Общая схема работы программы

Работа программы *SUMX*, а, следовательно, и блоков разделяется на три стадии. Во время первой стадии блоки вводят и анализируют ИПК. При анализе ИПК программа не прекращает работу после нахождения первой ошибки. Она накапливает неверно заданные ИПК и запоминает признак, который анализируется при переходе ко второй (операционной) стадии работы программы. Если признак равен 0, то продолжается выполнение программы, в противном случае печатается накопленная информация о неверно заданных ИПК и обрабатывается только 100 событий.

Во время второй стадии происходит считывание обрабатываемой информации с *DST* и выполняются операции, которые были определены во время первой стадии.

На третьей стадии производятся операции над гистограммами и вывод накопленных результатов из машины на печать или магнитную ленту.

Поскольку объем программы *SUMX* значительно превосходит размер оперативной памяти ЭВМ БЭСМ-6, отводимой для пользователя ( $\approx 20\text{К}$ ), то при запуске программы нам пришлось воспользоваться методом сегментации<sup>/4,5/</sup>. Его сущность заключается в том, что программа разбивается на сегменты (разделы), которые хранятся на магнитных барабанах и сменяют друг друга в оперативной памяти машины.

во время счета. В целях экономии машинного времени необходимо оптимально организовать разбиение программы на сегменты и, в частности, подпрограммы или функции, встречающиеся в сегментах одного уровня, выносить в головной сегмент. Отметим, что вторая стадия программы *SUMX*, которая является основной по затратам машинного времени при большом объеме обрабатываемого статистического материала, не сегментирована. Потеря времени из-за сегментации третьей стадии практически нет, так как последовательная смена сегментов в памяти ЭВМ происходит только один раз.

Если обрабатываемые данные (*DST*) находятся на магнитной ленте БЭСМ-6, то под гистограммы в *SUMX* отводится 10К слов. Если же обрабатываемые данные находятся на стандартной ленте фирмы *CDC*, то под гистограммы отводится около 7К слов, так как добавлены подпрограммы, преобразующие данные из формата *CDC* в формат БЭСМ-6.

## § 2. Мнемоническое задание номеров тестов и адресов величин на *DST* (задание словаря)

В *SUMX* имеется возможность мнемонического задания (в виде идентификаторов) номеров тестов и адресов величин на *DST* (блок *DICTIONARY*), что придает наглядность ИПК для блоков и особенно важно при большом объеме ИПК. Однако при распечатке ИПК на первой стадии блоки печатают только численные значения, соответствующие идентификаторам. Кроме того, использование идентификаторов вместо номеров тестов и адресов величин на *DST* позволяет пользователю изменять номера тестов и адреса величин на *DST*, изменяя только ИПК к блоку *DICTIONARY*.

На ИПК в первых 10-ти столбцах может быть целое число, уже определенный идентификатор или пробелы (для карт продолжения). Начиная с 11-го столбца задаются идентификаторы, разделяемые запятыми. Если слово на *DST* является целым числом, то перед идентификатором ставится знак минус ( $-JACK + 2$  означает  $-(JACK + 2)$ ).

Из любого идентификатора блока *DICTIONARY* может быть получено сколько угодно численных значений. Рассмотрим пример:

I	II
DICTIONARY	Y
27	JACK
501	TRACK/7

В этом случае идентификатору *JACK* присписывается значение  $I27$ , выражению (*JACK*+3) - значение  $I30$ , выражению (*JACK*-I2) - значение  $I15$  и т.д., а идентификатору *TRACK* - значение  $50I$ , выражению (*TRACK*+I00) - значение  $60I$  и т.д. Кроме того, наличие наклонной черты (/) и цифры (7) после *TRACK* указывает, что определяется матрица размерностью  $m \times n$ , где  $m$  - произвольное, а  $n=7$ . Это означает, что идентификатору *TRACK1* (*TRACK*, *TRACK1/I*) присписывается значение  $50I$ , *TRACK2* (*TRACK 2/I*) - значение  $508=50I+7$ , *TRACK3* (*TRACK 3/I*) - значение  $515=50I+2 \cdot 7$ , а *TRACK 3/6* присписывается  $520=50I+2 \cdot 7+(6-I)$  и т.д.

В приложении I дается пример ИПК для блока *DICTIONARY* и выдача программы *SUMX* для этого блока на первой стадии. В выдаче приводятся идентификаторы и присвоенные им числовые значения.

### § 3. Определение и использование тестов

Любая операция в программе *SUMX* может выполняться в зависимости от некоторого условия (теста). Операция совершается (например, событие заносится в гистограмму), если определяющий эту операцию тест удовлетворяется, и не совершается в противном случае.

Тесты определяются на ИПК для блока *SELECT* и идентифицируются номерами. Ранее определенный тест можно использовать при определении последующих тестов. При определении теста с номером *NT* автоматически определяется (неявно) тест *NT+I* как логическая противоположность теста *NT*. Однако тест *NT+I* в дальнейшем может быть явно определен. Явное определение теста *NT+I* должно быть в той же *EVA*-группе\*), в которой явно определяется тест *NT*, в противном случае на печать выдается специальный признак (\*PI ж, см. пример 2, приложение 2).

Не все тесты являются равноценными. Особое место занимает первый тест, который называется главным. Если он не удовлетворяется, то событие бракуется.

Значения всех определенных в блоке *SELECT* тестов вычисляются заново для каждого события и запоминаются в тест-векторе.  $n$ -ое слово тест-вектора содержит значение теста с номером  $n$ , которое является\*) Об *EVA*-группах см. далее.

ется отрицательным числом, если тест удовлетворяется, положительным числом, если тест не удовлетворяется, и нулем, если тест неопределен. Размеры тест-вектора определяются наибольшим номером из заданных тестов+1.

Объединение операций в *EVA*-группы дает возможность пропускать группы операций (например, гистограмм) в зависимости от выполняемости того или иного теста. Перед группой, которую можно пропускать, следует ставить карту *EVALUATE*, а в конце - карту *FINISH*. На карте *EVALUATE* в столбцах II+20 задается номер теста *NT*, от которого зависит пропуск. Такие объединения в *EVA*-группы применяются и в блоке *SELECT*. Поэтому тесты, попавшие в эту группу, будут определены, если тест *NT* удовлетворяется, и не определены - в противном случае.

### 3.1. Простые тесты

Простой тест задает отношение между величинами на *DST* и на ИПК. Используются следующие операции отношения:

- 1)  $C_1 < A < C_2$ , (*A BETWEEN*  $C_1$  и  $C_2$ );  $C_1, C_2$  - константы.
- 2)  $A < C$ ; (*A SMALLER*  $C$ );  $C$  - константа.
- 3)  $A > C$ , (*A BIGGER*  $C$ );
- 4)  $A = \begin{cases} \text{EQUAL} & C, \text{ применяется для сравнения целых чисел.} \\ \text{BCD} & C, \text{ применяется для сравнения величины на } DST \\ & \text{с BCD-константой на ИПК. Учитываются 6} \\ & \text{первых символов, включая пробелы, начиная с} \\ & \text{3I столбца.} \\ \text{ОСТАЛ} & C, \text{ применяется для сравнения величины на } DST \\ & \text{с восьмеричной константой на ИПК. Учитываются} \\ & \text{последние 12 символов.} \end{cases}$

Роль простого теста может выполнять величина на *DST*, указанная на ИПК при определении тестов или ранее определенный тест (например, 64-й тест, см. приложение 2).



### 3.2. Составные тесты

Пусть  $NT1, NT2, \dots, NTn$  — простые тесты. Тогда, соединив их между собой знаками "или" ( $OR$ ) или "и" ( $AND$ ) произвольным образом и расставив круглые скобки, мы получим составной тест

$NT = (NT1 \text{ или } NT2) \text{ и } NT3 \text{ и } (NT4 \text{ или } NT5 \text{ или } NT6)$ .

Этот составной тест удовлетворяется в том случае, если справедливы составляющие его тесты ( $NT1$  или  $NT2$ ),  $NT3$  ( $NT4$  или  $NT5$  или  $NT6$ ). Если хотя бы один из этих тестов не удовлетворяется, то и тест  $NT$  не удовлетворяется. "И" ( $AND$ ) на ИПК пробивается, а "ИЛИ" ( $OR$ ) подразумевается.

В примере I приложения 2 дается образец ИПК для блока *SELECT*. Знак минус перед номером теста указывает на то, что этот тест является величиной с *DST*. Всякий тест может быть позднее переопределен, и его новое определение может включать ранее сделанное определение. Например, 64-й тест переопределяется трижды, и его второе определение зависит от ранее определенных тестов 402 и 64.

### 3.3. Накопление списков

Обычно *SELECT* вычисляет количество событий (см. приложение 3), удовлетворяющих или не удовлетворяющих заданные тесты. Перфокарта *COUNT STOP* приостанавливает, а вновь появившаяся карта *COUNT* возобновляет этот процесс. Карты *LIST, LIST FAIL* дают возможность получать некоторую дополнительную информацию о событиях, удовлетворяющих или не удовлетворяющих данный тест. В примере 2 приложения 2 дается образец ИПК для блока *SELECT* и выдача программы *SUMX* на первой стадии. Здесь карта *LIST FAIL* генерирует список событий, не удовлетворяющих тест 310. Этот список идентифицируется номером теста (список *SELECT-310*). Знак минус указывает на то, что нужно накапливать информацию о событиях, не удовлетворяющих данный тест.

### § 4. Построение гистограмм и идеограмм (блок 6)

Этот блок накапливает и печатает в виде графиков гистограммы и идеограммы величин, расположенных на *DST* с соответствующими ве-

сами. Блок строит идеограмму в том случае, если на ИПК задана среднеквадратичная ошибка. Гистограммы и идеограммы идентифицируются их порядковыми номерами. В связи с этим иногда необходимо вводить фиктивные гистограммы. Так, при обработке карты

I	II
DUMMY	12

блок 6 генерирует  $n$  фиктивных гистограмм, которые не выдаются на печать.

Отметим, что

- 1) Данное событие может давать несколько вкладов в одну гистограмму. Каждый вклад обусловлен соответствующим тестом (в приложении 4 в I и 2-ю гистограммы - 3 вклада, а в I8-ю - четыре вклада).
- 2) Для группы гистограммы может быть задан один или несколько определяющих тестов с помощью карт *EVAL* и *FINISH*. Данное событие будет давать вклад в гистограммы группы, если удовлетворяются эти тесты (в приложении 4 вклад в I и 2 гистограммы зависит от тестов I00 и I02, а в I8-ю - от тестов I00 и I03).

Для каждой гистограммы на печать выдается (см. приложение 4):

- 1) График гистограммы. Высота каждого столбца на графике представляет содержимое канала. Каждый знак "x" в столбце ставится для величины, равной приращению вертикальной шкалы. Верхним символом столбца может быть число, меньшее I0 или буква M, если число  $\geq$  I0, но меньше величины, необходимой для печати очередного знака "x".
- 2) Строка *SIGN*, которая содержит минус для каждого канала с отрицательным содержанием.
- 3) Содержимое каждого канала по вертикали (*CONTENTS* ).
- 4) Номера каналов (*CHAN. NOS.* ).
- 5) Значения абсцисс левых границ каналов. Для отрицательных абсцисс сверху ставится знак минус.
- 6) Сумма содержимого всех каналов, встречающихся на графике (*CONTENTS ALL CHAN.* ).
- 7) Содержимое каналов, выходящих за пределы гистограммы и число входов в идеограмму с весовым множителем.

§ 5. Подготовка массива перфокарт для обсчета числового материала с МЛ БЭСМ-6

Рабочий вариант программы находится на МЛ. Для работы должна быть подготовлена следующая колода перфокарт:

I	II	2I	3I
*NAME			
*ASSIGN $\downarrow$ LT APE			
*ASSIGN $\downarrow$ FTAPE $\downarrow$ 12			
*ASSIGN $\downarrow$ FTAPE $\downarrow$ 13			
*ASSIGN $\downarrow$ FTAPE $\downarrow$ 05			
*NO $\downarrow$ LOAD $\downarrow$ LIST			
*PERSONAL $\downarrow$ LIBRARY			
Подпрограммы	CHARML		
*CALL $\downarrow$ FULL MEMORY			
*EXECUTE			
ИПК			
*END FILE			
Диспетчерский	конец		

Логические номера рабочих лент, задаваемые на картах \*ASSIGN FTAPE, начинаются с I2, отличаются друг от друга на 1 (I2, I3, ...) и должны совпадать с номерами, заданными на карте \*LIST. Логические номера DST также задаются на картах \*ASSIGN FTAPE, начиная с 05, отличаются друг от друга на 1 и должны совпадать с номерами, заданными на картах \*TAPE.

§ 6. Подготовка массива перфокарт для обсчета числового материала с МЛ фирмы СДС

Для работы должна быть составлена колода из следующих перфокарт:

I	II	2I	3I
*NAME			

*ASSIGN	LT	TAPE		
*ASSIGN	FTAPE	05		
*ASSIGN	FTAPE	12		
*ASSIGN	FTAPE	13		
*NO LOAD	LIST			
*PERSONAL	LIBRARY			
Подпрограммы:	RD CDC , CDC	*606* , ERR CDC } RBESH , IBESH CRBESH	} для работы с МЛ CDC	
Подпрограммы:	CHARM: REDST1 } SUM34 }		для обсчета DST CDC	
*CALL	FULL	MEMORY		
*EXECUTE				
	ИПК			
*END FILE				
Диспетчерский	конец			

### § 8. Некоторые возможности программы

1) Если необходимо составить списки, используя только блок SELECT , можно поставить карту

I	II
*GØ ANYWAY	

перед \*ALL DONE (последняя карта ИПК).

2) Если не нужно выдавать на печать информацию об ИПК на первой стадии, то

I	II
*OUTPUT	I2

надо подложить за \*NEW PASS , а карту \*OUTPUT - перед \*ALL DONE . Тогда результаты будут выдаваться на МЛ с логическим номером I2.

3) Если нужно увеличить память под программы CHARM: , то надо уменьшить память, отводимую под гистограммы. Для этого в главной программе SUM34 надо изменить две перфокарты

COMMON TEMP (500), CM (10000) / CHIEF / ITA , ITB , ITPU , \*STAGE, IROOM, ITEMS  
IIMAX = 10000

В заключение авторы выражают благодарность Н.Н.Говоруну, В.Г.Иванову, И.С.Саитову за помощь в работе.

#### ЛИТЕРАТУРА

1. TC Program Library, CERN, v. 1, 2, 3, 1968.
2. А.Ф.Лукьянцев, И.С.Саитов. "Использование программы SLICE на CDC-1604A для экспериментов со 100-см водородной пузырьковой камерой ЛВЭ ОИЯИ".  
Сообщение ОИЯИ, P10-5210, Дубна, 1970.
3. J. Zoll „Program SUMX“.  
Препринт ОИЯИ, IO-4319. Дубна, 1968.
4. Н.А.Буздавина, Н.Н.Говорун, Л.Дорж, А.Г.Заикина, В.Г.Иванов, Л.И.Лепилова, А.Ф.Лукьянцев, В.В.Макеев, Б.А.Манюков.  
"Система программ обработки фильмовой информации с жидководородных камер ОИЯИ на ЭВМ БЭСМ-6". Сообщение ОИЯИ, P10-5785, Дубна, 1971.
5. И.Н.Силин, В.Ю.Веретенев, Н.С.Заикин. "Мониторная система БЭСМ-6. Общая организация". Материалы совещания по программированию и вычислительным методам решения физических задач, II-4655, ч. I, Дубна, 1969.

Рукопись поступила в издательский отдел

28 июля 1971 года.

ПРИЛОЖЕНИЕ I

I. Образец ИПК для блока DICTIONARY.

Столбцы	11	21	31	41
*DICTIONARY	Y			
I	.EXP, EVENT,	NC, DATE, TI	ME, CLASS, S	SERIAL
IO	BASE			
BASE	NUM+202, TH	OV, TENTHO		
BASE+2	LINE 5/2			
I50I	SPBOUT			
I00	GROUP3/100			
2I	TT/2			
TT/2	TF/2			
5I	FF/2			
FF/2	FT/2			
IO	X			
35	W			
35	W			
52	MIN			
II2	MAX			
I36	UNDEF			

2. Распечатка ИПК для блока DICTIONARY на начальной стадии

-----SUM 466 STANDART TEST RUN-----+PASS 1-1 .... PAGE 1-1  
 REQUESTED ----- \* DICTIONARY -47  
 DICTIO HAS CONTROL

I = EXP	I/	I
2 = EVENT	I/	I
3 = NC	I/	I
4 = DATE	I/	I
5 = TIME	I/	T
6 = CLASS	I/	I
7 = SERIAL	I/	I
IO = BASE	I/	I
IO = NUM	I/	203
2I3 = THOU	I/	I
2I4 = TENTHO	I/	I
I2 = LINE	5/	2
I50I = SPBOUT	I/	I
I00 = GROUP	3/	100
2I = TT	I/	2
22 = TF	I/	2
5I = FF	I/	2
52 = FT	I/	2
IO = X	I/	I
35 = W	I/	I
35 = W	I/	I
52 = MIN	I/	I
II2 = MAX	I/	I
I36 = UNDEF	I/	I

RE-DEFINED 'W'

ПРИЛОЖЕНИЕ 2

1. Образец ИПК для блока SELECT

Столбцы	11	21	31	41	
*SELECT TEST	410 15	BETWEEN	7395.64	9758.5	Тест с открытым интервалом
TEST	402 -51 55 81	EQUAL TO BCD OCTAL	753I	123400	Точное сравнение BCD-сравнение Восьмеричное сравнение
TEST	64 59 59	EQ BET FAIL	3I.35 3I.3495	3I.3505	Очень опасно Это надежная запись
LIST EVAL TEST	410 64 402 64	TRUE FALSE			Пропуск в SELECTE Переопределение теста
FINISH EVAL TEST	41I 64 402 64 73	FALSE FAL FAL			Неявно определенный тест
FINISH TEST	I I 3	EQU TRUE	86		Главный тест
AND AND LIST	64 -124 -175	TRU FAL			т.е.124 слово DST отриц. или ноль. т.е.175 слово DST положит.

2. Образец ИПК для блока SELECT

Столбцы	11	21	31	41	
*SELECT EVAL TEST	GROUP3 GROUP3 MAX	BET	9.9	22.01	FIRST INTERVAL
TEST	GROUP3/5 MAX	BET	9.9	18.01	SECOND INT
TEST	GROUP3/7 MAX	BET	9.9	16.01	THIRD INT
TEST	GROUP3/9 MAX	BET	9.9	14.01	FOURTH INT
TEST	GROUP3/11 MAX	BET FAIL STOP	9.9	12.01	FIFTH INT
LIST COUNT TEST	1.01 1	BCD	DUMMY		
TEST	103 1	BCD	DUMMY		
TEST	322 1	BCD	DUMMY		
COUNT TEST	320 12	BIG	18		

2.1. Образец распечатки ИМК для блока SELECT на начальной стадии.

----- SUNY 466 STANDARD TEST RUN ----- PAGE 6 -6  
 REQUESTED ----- \*SELECT  
 SELECT HAS CONTROL BLOCK 5

COUNTING STARTS  
 \*\* \*\*\*\*\* LEVEL 1 TEST 300

TEST 302 TRUE IF WORD 112 BETWEEN 9.9000 AND 22.0100 FIRST INTERVAL

TEST 304 TRUE IF WORD 112 BETWEEN 9.9000 AND 18.0100 SECOND INT

TEST 306 TRUE IF WORD 112 BETWEEN 9.9000 AND 16.0100 THIRD INT

TEST 308 TRUE IF WORD 112 BETWEEN 9.9000 AND 14.0100 FOURTH INT

TEST 310 TRUE IF WORD 112 BETWEEN 9.9000 AND 12.0100 FIFTH INT

LIST FAILING EVENTS.

COUNTING STOPS.

TEST 101 TRUE IF

WORD 1 HAS BCD 'DUMMY'

TEST 103 TRUE IF

WORD 1 HAS BCD 'DUMMY'

TEST 322 TRUE IF

WORD 1 HAS BCD 'DUMMY'

COUNTING STARTS

TEST 320 TRUE IF

WORD 12 BIGGER THAN WORD 18 + 0.0000

----- LEVEL 0 FINISHED -----

SUMMARY OF EXPLICITLY DEFINED TESTS.

1	20	21	23	25	27	29	31	50	51	53	55	57	59	60	61	100	101	102	103
104	106	108	300	302	304	306	308	310	320	322									
STORE	1995	308E		3549	3549	3549	3549	3549	3549	3549	3549	3549	3549	3549	3549	3549	3549	3549	3549

7328 WORDS UNUSED



Список событий, не удовлетворяющих тест 310

--- SUMX 466 STANDART TEST RUN --- +PASS 1-1.....PAGE 20-20  
SELECT/-310 POOL 2 PG.1

RECORD EXP ROLL /FRAME GRIND DATE -TIME

37	99	25/I036.I	66030I	I
38	99	25/I037.I	66030I	2
39	99	25/I038.I	66030I	3
40	99	25/I039.I	66030I	4
41	99	25/I040.I	66030I	5
42	99	25/I041.I	66030I	6
43	99	25/I042.I	66030I	7
44	99	25/I043.I	66030I	8
45	99	25/I044.I	66030I	9
46	99	25/I045.I	66030I	10
47	99	25/I046.I	66030I	11
48	99	25/I047.I	66030I	12
49	99	25/I048.I	66030I	13
50	99	25/I049.I	66030I	14
51	99	25/I050.I	66030I	15
52	99	25/I051.I	66030I	16
53	99	25/I052.I	66030I	17
54	99	25/I053.I	66030I	18
55	99	25/I054.I	66030I	19
56	99	25/I055.I	66030I	20
I08	99	25/II07.I	66030I	72
I09	99	25/II08.I	66030I	73
III	99	25/IIII.I	66030I	75
II2	99	25/IIII.I	66030I	76
II3	99	25/IIII.I	66030I	77
II4	99	25/IIII.I	66030I	78
II5	99	25/IIII.I	66030I	79
II7	99	25/IIII.I	66030I	81
II9	99	25/IIII.I	66030I	83
I20	99	25/IIII.I	66030I	84

30 ENTRIES. INCL. 0 RTT. 0 FAULTY RECORDS ON BUFFER TAPE 12  
CHAR9 HAS BEEN CALLED

ПРИЛОЖЕНИЕ 3

Список кол-ва событий, удовлетворяющих и не удовлетворяющих данные тесты

--- SUMX 466 STANDART TEST RUN --- +PASS 1-1 .... PAGE 19-19  
SELECT BLOCK-5 NUMBER OF EVENTS WHICH PASSED OR FAILED THE TESTS

TEST	PASSED	FAILED	SUM	MASTER	TESTS
302	96	0	96	300	
304	84	12	96	300	
306	74	22	96	300	
308	63	33	96	300	
310	50	46	96	300	
COUNTING	MODE	CHANGED			
101	0	0	0	300	
103	0	0	0	300	
322	0	0	0	300	
COUNTING	MODE	CHANGED			
320	37	59	96	300	

ПРОТОКОЛ № 4.

----- SUMX 466 STANDARD TEST-RUN  
 REQUESTED -----  
 BLOCK 6 HAS CONTROL

\*\*\*\*\*  
 \*\* 1 \*\*\*\*\*  
 \*\* 2 \*\*\*\*\*  
 PROGRAM 1 13 ENTRIES IDEOGRAMMED -- WRITE USE OF SPECIAL 80UT OPT.  
 M-D-L, PT, F, I, U, LG 80 1.0000 0.0000 0 1000.0000 0PT. OPT.  
 X-W-T-S-LOC, SCALE 14 0 0 1501 1.0000 TEST ...  
 X-W-T-S-LOC, SCALE 30 0 0 1501 1.0000  
 X-W-T-S-LOC, SCALE 86 0 0 1502 4.0000  
 STORE 2372 DONE DYN ST 9262, 9549, 9549, 9849, 9849, 10000 6937 WORDS UNUSED

PROGRAM 2 1 SAME AS LAST, WITH LOWER LIMIT OPTION  
 M-D-L, PT, F, I, U, LG 80 1.0000 0.0000 0 1000.0000 OPT. YES OPT.  
 X-W-T-S-LOC, SCALE 14 0 0 1501 1.0000  
 X-W-T-S-LOC, SCALE 50 0 0 1501 1.0000  
 X-W-T-S-LOC, SCALE 86 0 0 1501 1.0000  
 STORE 2372 DONE DYN ST 9262, 9549, 9549, 9849, 9849, 10000 6790 WORDS UNUSED

PROGRAM 3 1 DUMMY 15  
 STORE 2679 DONE DYN ST 9262, 9549, 9549, 9849, 9849, 10000 6760 WORDS UNUSED  
 -----  
 \*\* 1 \*\*\*\*\*  
 \*\* 2 \*\*\*\*\*

PROGRAM 18 HISTOGRAM SINCE NO SIGMA-LOC SPECIFIED  
 M-D-L, PT, F, I, U, LG 141 1.0000 -10.2500 0 1.0000 OPT. OPT.  
 X-W-T-S-LOC, SCALE 10 35 0 0 1.0000  
 X-W-T-S-LOC, SCALE 14 36 0 0 1.0000  
 X-W-T-S-LOC, SCALE 42 37 0 0 1.0000  
 X-W-T-S-LOC, SCALE 13 38 0 0 1.0000  
 X-W-T-S-LOC, SCALE 14 30 196 0 1.0000  
 X-W-T-S-LOC, SCALE 15 40 196 0 1.0000  
 X-W-T-S-LOC, SCALE 16 41 196 0 1.0000  
 X-W-T-S-LOC, SCALE 17 42 196 0 1.0000  
 X-W-T-S-LOC, SCALE 18 43 196 0 1.0000  
 X-W-T-S-LOC, SCALE 19 44 193 0 1.0000  
 X-W-T-S-LOC, SCALE 20 45 103 0 1.0000  
 X-W-T-S-LOC, SCALE 31 46 103 0 1.0000  
 STORE 2713 DONE DYN ST 9262, 9549, 9549, 9849, 9849, 10000 6854 WORDS UNUSED  
 -----  
 \*\* 1 \*\*\*\*\*  
 \*\* 2 \*\*\*\*\*

NOT DEFINED  
 NOT DEFINED  
 NOT DEFINED  
 NOT DEFINED

