Сёмкин Арсений Сергеевич, БПИ-237

**Архитектура вычислительных систем, ИДЗ-1, вариант 27**

1. **Требования к ДЗ**

Разработать программу, которая вводит одномерный массив A, состоящий из N элементов (значение N вводится при выполнении программы), после чего формирует из элементов массива A новый массив B по правилам, указанным в варианте, и выводит его. Память под массивы может выделяться статически, на стеке, автоматически по выбору разработчика с учетом требований к оценке работы.

При решении задачи необходимо использовать подпрограммы для реализации ввода, вывода и формирования нового массива. Допустимы (при необходимости) дополнительные подпрограммы.

Максимальное количество элементов в массиве не должно превышать

10 (ограничение обуславливается вводом данных с клавиатуры). При этом необходимо обрабатывать некорректные значения как для нижней, так и для верхней границ массивов в зависимости от условия задачи.

Вариант 27: cформировать массив B из элементов массива A. Элементы массива, оканчивающиеся цифрой 4, уменьшить вдвое.

1. **Критерии на 4-5**

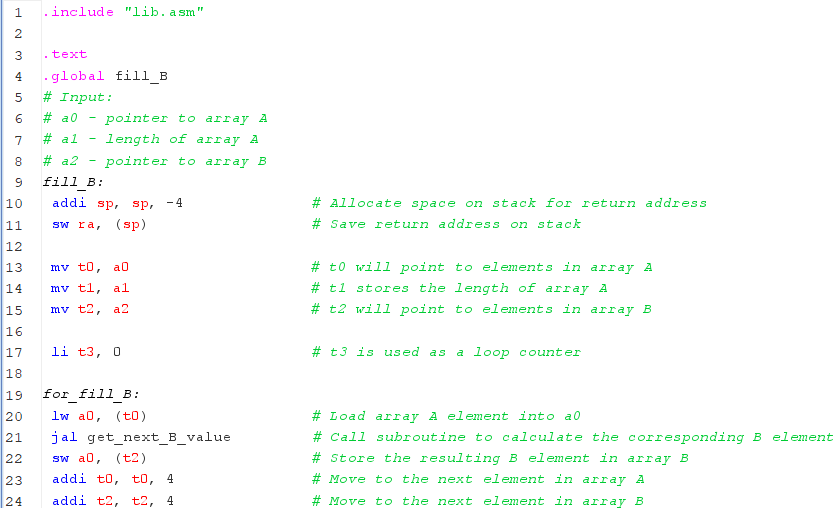
• **Приведено решение задачи на ассемблере. Ввод данных осуществ-**

**ляется с клавиатуры. Вывод данных осуществляется на дисплей.**

Выполнено.

• **В программе должны присутствовать комментарии, поясняющие**

**выполняемые действия.**

Их очень много: 

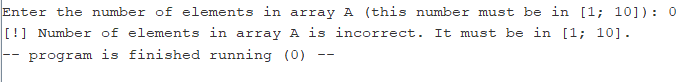
• **Допускается использование требуемых подпрограмм без параметров и локальных переменных.**

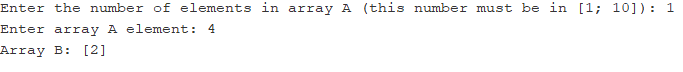
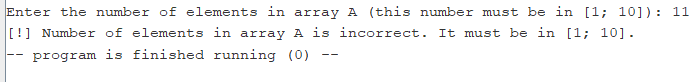
Поскольку я выполнял критерии на 10 баллов, подпрограммы получают и возвращают данные согласно конвенции через регистры a. Локальные переменные всегда находятся в t в подпрограммах, могут быть в s в главной программе. Более подробно будет раскрыто далее.

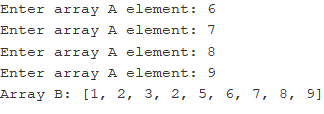
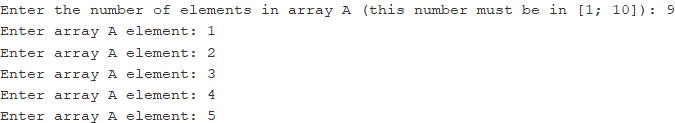
• **В отчете должно быть представлено полное тестовое покрытие.**

**Приведены результаты тестовых прогонов. Например, с использованием скриншотов.**

Были совершены различные тестовые прогоны. Более детально это будет описано в разделе про автоматическое тестирование. Здесь вижу смысл отразить тесты с различными размерами массивов.







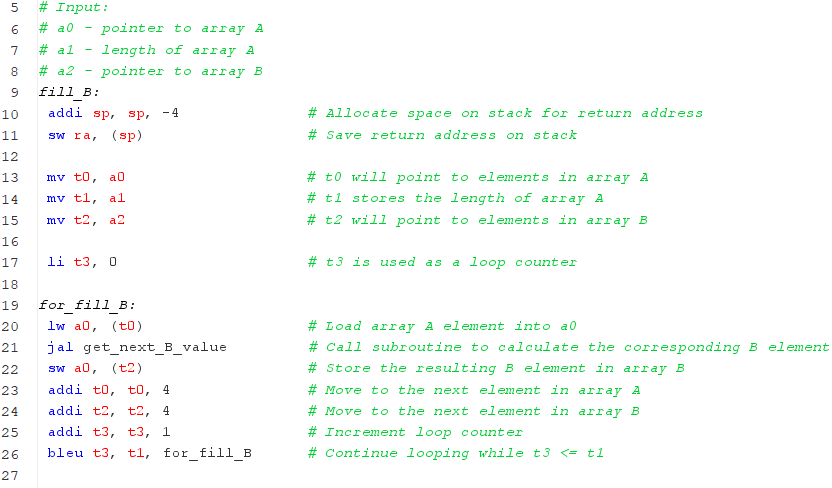
Как видим, программа успешно обрабатывает некорректные размеры массива.

1. **Критерии на 6-7**

• **В программе необходимо использовать подпрограммы с передачей**

**аргументов через соответствующие регистры, определяемые конвенцией по их использованию. При нехватке этих регистров для их хранения необходимо использовать стек.**

Как уже было описано выше, все регистры используются согласно конвенции. В программе используются подпрограммы, они получают аргументы и возвращают результат через соответствующие регистры a.



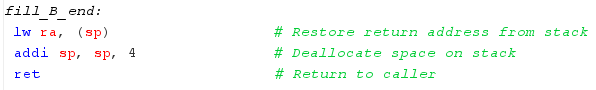
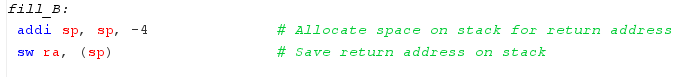
Например, вот часть подпрограммы, которая заполняет массив B. Она принимает указатель на начало массива A через a0, длину массива А через а1 и указатель на начало B через а2. Соответствующим образом построены и другие подпрограммы, а их достаточно много.

**• Локальные переменные допускается размещать в свободных реги-**

**стров, при нехватке которых необходимо их сохранять на стеке.**

**Или сразу все локальные переменные отображать на стек.**

Все локальные переменные в подпрограммах сохраняются в регистрах t. Не пришлось сохранять их на стеке, потому что всегда хватало регистров t. Однако в подпрограмме, заполняющей массив B пришлось сильно подумать, как использовать эти регистры, чтобы их хватило и не пришлось выделять память для переменных на стеке. Также стоит добавить, что на стеке сохраняются адреса возврата для подпрограмм, которые внутри себя так же вызывают подпрограммы. Когда подпрограмма завершает работу, со стека подгружается адрес возврата, затем он очищается со стека и происходит вовзрат к месту, откуда подпрограмма была вызвана.



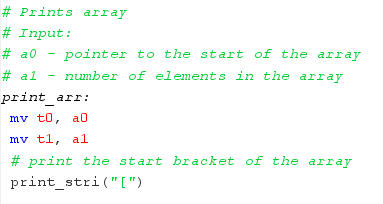
**• В местах вызова функции добавить комментарии, описывающие пе-**

**редачу фактических параметров и перенос возвращаемого резуль-**

**тата. При этом необходимо отметить, в каких регистрах или где на**

**стеке отображаются соответствующие фактические параметры.**

Пример был выше, вот еще пример:



**• Информацию о проведенных изменениях отобразить в отчете на-**

**ряду с информацией, необходимой на предыдущую оценку.**

Выполнено.

1. **Критерии на 8**

• **Разработанные подпрограммы должны поддерживать многократ-**

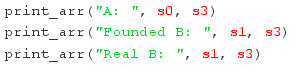
**ное использование с различными наборами исходных данных, вклю-**

**чая возможность подключения различных исходных и результиру-**

**ющих массивов. То есть, поддерживать работу с формальными и**

**фактическими параметрами.**

С учетом использования подпрограмм и макросов с разными наборами исходных данных удалось избежать дублирования кода. Например выводы строк или массивов:







**• Реализовать автоматизированное тестирование за счет создания**

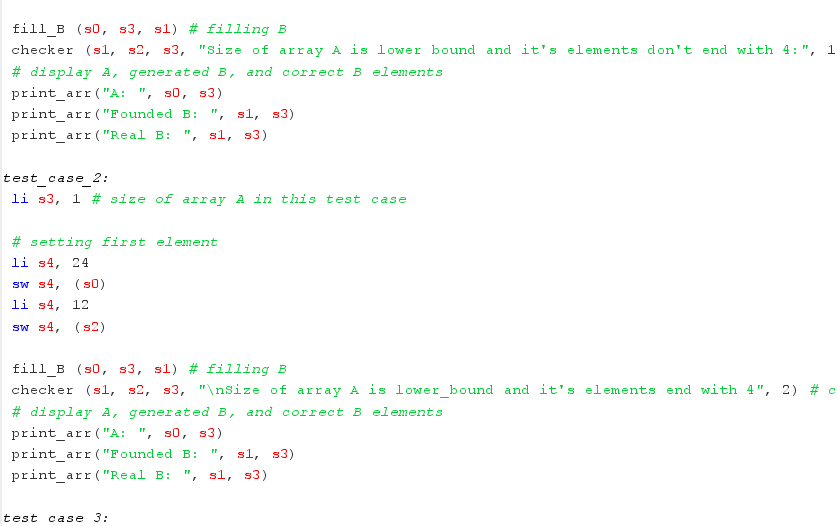
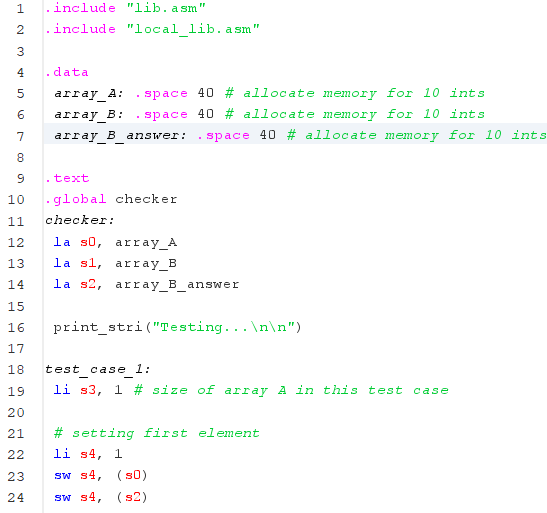
**дополнительной тестовой программы, осуществляющей про-**

**гон подпрограммы обработки массивов с различными тестовыми**

**данными (вместо ввода данных). Осуществить прогон тестов обес-**

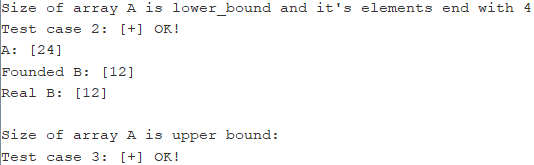
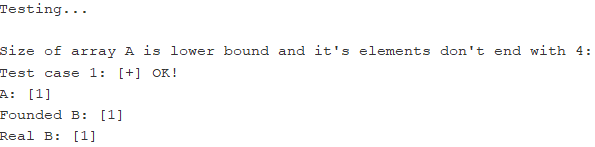
**печивающих покрытие различных ситуаций. Тестовые данные мож-**

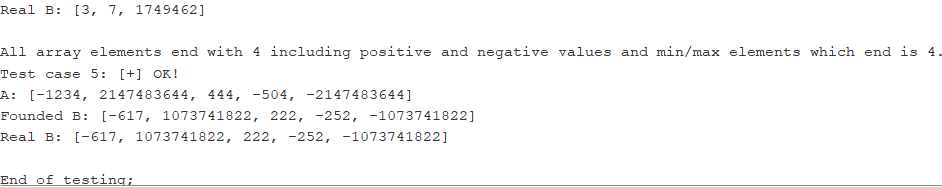
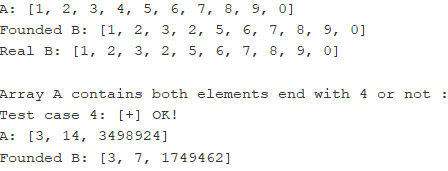
**но формировать в различных исходных массивах.**

Был добавлен файл checker.asm, содержащий программу осуществляющую автоматическое тестирование на различных наборах входных данных.

Протестированы различные наборы входных данных:

* Минимальный/максимальный размер массива
* Массив состоящий только из чисел, не заканчивающихся на 4
* Массив состоящий только из чисел, заканчивающихся на 4
* Массив состоящие и из чисел, заканчивающихся на 4 и из чисел не заканчивающихся на 4.
* Массив состоящий из чисел разных знаков, в том числе чисел, заканчивающихся на 4 разных знаков (что важно для логики проверки, заканчивается ли число на 4 для отрицательных чисел)
* Массив включает максимально и минимально возможные числа, заканчивающиеся на 4.





**• Добавить информацию о проведенных изменениях в отчет.**

Выполнено.

1. **Критерии на 9 баллов**

**• Добавить в программу использование макросов которые должны**

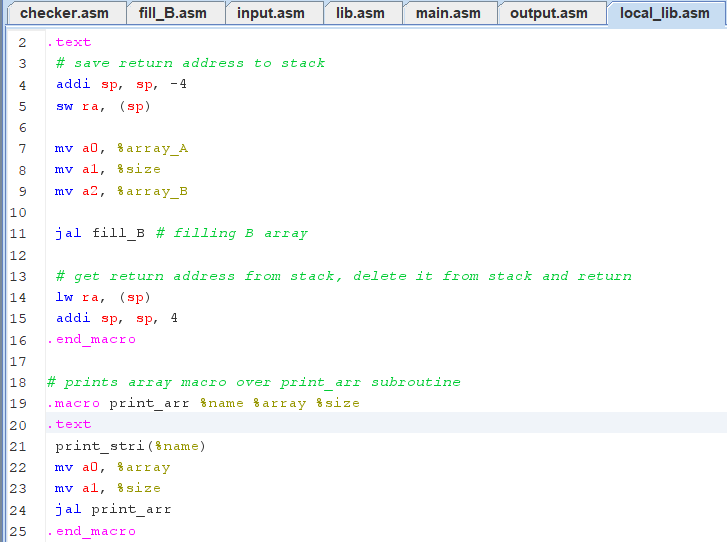
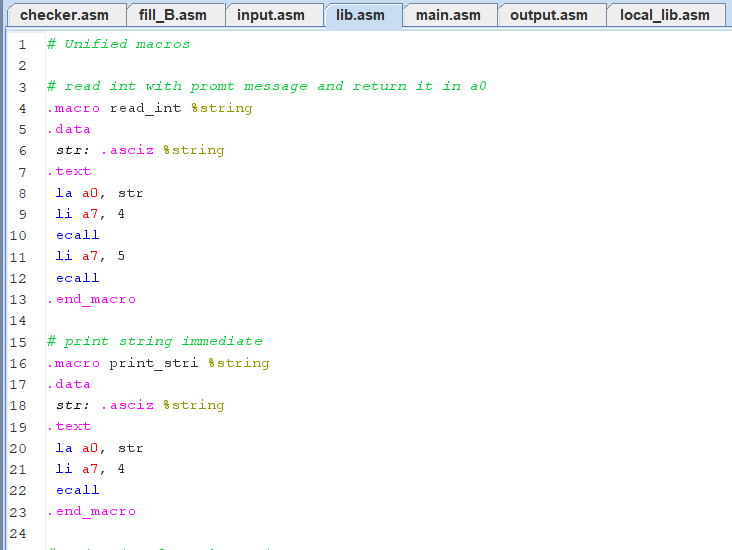
**использоваться в качестве оберток ранее написанных подпрограмм**

**ввода и вывода данных, генерации тестовых массивов. Макросы**

**должны поддерживать повторное использование с различными мас-**

**сивами и другими необходимыми параметрами.**

Макросов добавлено достаточно много. lib.asm содержит макросы, которые можно использовать вне этой программы. local\_lib.asm содержит макросы, которые можно использовать только внутри этой программы.



**• Допускается реализация в виде собственных макросов и других**

**фрагментов программы.**

Как описано выше, собственные макросы в local\_lib.asm, а несобственные в lib.asm

1. **Критерии на 10**

**• Программа должна быть разбита на несколько единиц компиляции**

**(ассемблерных файлов). При этом подпрограммы ввода–вывода долж-**

**ны составлять унифицированные модули, используемые повторно**

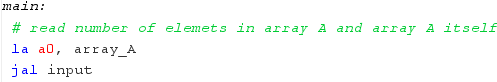
**как в программе, осуществляющей ввод или генерацию исходных**

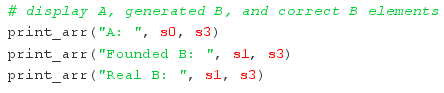
**данных, так и в программе, осуществляющей тестовое покрытие.**

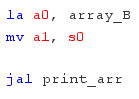
Программа разбита на несколько единиц компиляции:



Подпрограммы ввода-вывода используются в основной программе и в программе реализующей тестирование.







**• Макросы должны быть выделены в отдельную автономную библиотеку**

Как описано выше, собственные макросы в local\_lib.asm, а несобственные в lib.asm.

**• Расширить отчет, дополнив его новыми данными.**

Выполнил.