

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
"НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
"ВЫСШАЯ ШКОЛА ЭКОНОМИКИ"

Факультет компьютерных наук
Департамент программной инженерии

ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ № 1 ПО ДИСЦИПЛИНЕ
АРХИТЕКТУРА КОМПЬЮТЕРА И ОПЕРАЦИОННЫЕ СИСТЕМЫ

Москва 2025

АННОТАЦИЯ

Материал содержит задания по дисциплине «Архитектура компьютера и операционные системы», предназначенные для самоподготовки студентов, обучающихся на образовательной программе 09.03.04 «Программная инженерия», реализуемой ФКН НИУ ВШЭ. Наряду с заданиями представлены требования по их выполнению.

СПИСОК СОКРАЩЕНИЙ

- API — Application Programming Interface. Интерфейс Программирования приложения — программный интерфейс, то есть описание способов взаимодействия одной компьютерной программы с другими.
- ISA — Instruction Set Architecture
- POSIX — Portable Operating System Interface. Переносимый интерфейс операционных систем — набор стандартов, описывающих интерфейсы между операционной системой и прикладной программой (системный API), библиотеку языка C и набор приложений и их интерфейсов.
- RARS — RISC-V Assembler and Runtime Simulator. Ассемблер и имитатор, выполняющий программы на языке ассемблера RISC-V. Его основная цель — стать эффективной средой разработки для людей, начинающих работать с RISC-V.
- RISC — Reduced Instruction Set Computer
- RISC-V — расширяемая открытая и свободная система команд и процессорная архитектура на основе концепции RISC
- ОС — операционная система
- ЯП — язык программирования
- ЯПВУ — язык программирования высокого уровня

Содержание

1	Задание № 1	5
1.1	Варианты заданий	5
2	Требование к выполнению и оценка задания № 1	9

1 Задание № 1

Разработать программу, в которой вводится одномерный массив A , состоящий из N элементов (значение N вводится в диалоге с пользователем при выполнении программы). После этого из элементов массива A формируется новый массив B в соответствии с условиями, указанными в варианте, и осуществляется его вывод. 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 массива. Допустимы (при необходимости) дополнительные подпрограммы.

Максимальное количество элементов в массиве не должно превышать 10 (ограничение должно контролироваться при вводе данных). Необходимо обрабатывать некорректные значения как для нижней, так и для верхней границ массивов в зависимости от условия задачи.

1.1 Варианты заданий

1. Сформировать массив B из положительных элементов массива .
2. Сформировать массив B только из тех элементов массива , которые не совпадают с его первым и последним элементами.
3. Сформировать массив B из сумм соседних элементов A по следующим правилам:

$$B_0 = A_0 + A_1, B_1 = A_1 + A_2, \dots$$
4. Массив B из массива A формируется по следующим правилам:
 - $B_i = 1$, если $A_i > 0$,
 - $B_i = -1$, если $A_i < 0$,
 - $B_i = 0$, если $A_i = 0$.
5. Сформировать массив B , состоящий из элементов массива A , значение которых не совпадает с введённым числом X .
6. Сформировать массив B , состоящий из элементов массива A , значения которых кратны введённому числу X .
7. Сформировать массив B из индексов положительных элементов массива A .
8. Сформировать массив B по следующим правилам:
 - если $A_i > 5$, то увеличить элемент на 5,

- если $A_i < -5$, то уменьшить на 5,
- остальные обнулить.

9. Сформировать массив B из нечётных элементов массива A .
10. Сформировать массив B из отрицательных элементов массива A , расположив их обратном порядке.
11. Сформировать массив B из элементов A , расположенных в обратном порядке, исключая из нового массива первый положительный элемент.
12. Сформировать массив B из элементов массива A , исключив первый положительный и последний отрицательный элементы.
13. Сформировать массив B из элементов массива A , за исключением элементов, значения которых совпадают с минимальным элементом массива A .
14. Сформировать массив B из элементов массива A заменой всех отрицательных значений на максимум из массива A .
15. Сформировать массив B из элементов массива A заменой всех нулевых элементов значением минимального элемента.
16. Сформировать массив B из элементов массива A , заменой значений, которые больше среднего арифметического, значением среднего арифметического массива A .
17. Сформировать массив B из элементов массива A , расположенных после последнего положительного элемента.
18. Сформировать массив B из элементов массива A уменьшением на 5 всех элементов, встретившихся до первого положительного.
19. Сформировать массив B из элементов массива A заменой нулевых элементов, предшествующих первому отрицательному, единицей.
20. Сформировать массив B из элементов массива A перестановкой местами минимального и первого элементов.
21. Сформировать отсортированный по возрастанию массив B из элементов массива A .
22. Сформировать отсортированный по убыванию массив B из элементов массива A .
23. Сформировать массив B , элементы которого являются числами Фибоначчи от соответствующих элементов массива A . В случае переполнения в соответствующие места записывать нулевые значения.

24. Сформировать массив B из элементов массива A поменяв местами элементы, стоящие на чётных и нечётных местах:
 $A_0 \leftrightarrow A_1; A_2 \leftrightarrow A_3 \dots$
25. Сформировать массив B из элементов массива A заменив все положительные числа значением 2, а отрицательные — увеличить на 5.
26. Сформировать массив B из сумм трех соседних элементов массива A , где $B_i = A_{i-1} + A_i + A_{i+1}$. Если элементов в массиве A менее трёх, то заполнить массив B нулями.
27. Сформировать массив B из элементов массива A . Элементы массива A , оканчивающиеся цифрой 4, уменьшить вдвое.
28. Сформировать массив B из тех элементов массива A , которые образуют неубывающие последовательности. Неубывающей последовательностью считать элементы идущие подряд, которые равны между собой или каждый последующий больше предыдущего.
29. Сформировать массив B из сумм соседних элементов A по следующему правилу:
 $B_0 = A_0, B_1 = A_0 + A_1, \dots, B_m = A_0 + \dots + A_m$, где m — либо номер первого четного отрицательного элемента массива, либо номер последнего элемента, если в массиве нет отрицательных элементов.
30. Сформировать массив B из элементов массива A , которые больше, чем элементы, стоящие перед ними.
31. Сформировать массив B из элементов массива A в следующем порядке: сначала заполняем массив числами, стоящими на нечетных местах, а затем — стоящими на четных местах в массиве.
32. Сформировать массив B из элементов массива A , которые меньше суммы элементов, расположенных на четных местах.
33. Сформировать массив B из элементов массива A , путем записи разности между двумя соседним элементами.
34. Сформировать массив B из элементов массива A заменив элементы на четных местах суммой всех положительных элементов, а элементы на нечетных местах суммой отрицательных элементов.
35. Сформировать массив B из элементов массива A сгруппировать положительные элементы массива в начале массива, нулевые в середине, а отрицательные — в конце.

36. Сформировать массив B из элементов массива A сгруппировав элементы с четными индексами в начале массива , а элементы с нечетными индексами сгруппировать в конце массива B .
37. Сформировать массив B из элементов массива A в следующем порядке: элементы с индексами $i \leq (N + 1)/2$ переместить на позиции с четными индексами массива с сохранением их исходного порядка относительно друг друга, а оставшиеся элементы ($i > (N + 1)/2$) разместить на позициях с нечетными индексами массива также с сохранением их исходного порядка.
38. Сформировать массив B из элементов массива A , которые одновременно имеют четные и отрицательные значения.
39. Сформировать массив B , элементы которого являются площадью квадратов со сторонами указанные в массиве A . Операцию умножения реализовать через сложение и/или сдвиги. При переполнении в массив B записывать нули.
40. Сформировать массив B из сумм соседних элементов A по следующему правилу: $B_0 = A_0, B_1 = A_0 + A_1, \dots, B_m = A_0 + \dots + A_m$, где m – номер первого элемента массива A большего среднего арифметического этого массива. При переполнении записывать нули.
41. Сформировать массив B из индексов элементов массива A . Порядок следования индексов в массиве B позволяет выводить элементы массива A по возрастанию.
42. Сформировать массив B из индексов элементов массива A . Порядок следования индексов в массиве B позволяет выводить элементы массива A по убыванию.
43. Сформировать массив B , элементы которого являются факториалами от соответствующих элементов массива A . В случае переполнения в соответствующие места записывать нулевые значения.

2 Требование к выполнению и оценка задания № 1

Ниже представлены требования, которым должно удовлетворять задание по разработке программы на ассемблере для получения соответствующей оценки.

Следует отметить, что необходимо представить только одну ассемблерную программу, разработанную на целевую оценку.

Для обеспечения лучшей переносимости кода предлагается при работе с текстовыми сообщениями использовать английский язык или транслит. То есть, пользоваться только кодировкой ASCII. Но это не является обязательным требованием.

4–5 баллов

- Приведено решение задачи на ассемблере. Ввод данных осуществляется с клавиатуры. Вывод данных осуществляется на дисплей.
- Программа в ходе работы должна выводить в консоль как исходный так и сформированные массивы, сопровождая вывод каждого из них соответствующими пояснениями.
- В программе должны присутствовать комментарии, поясняющие выполняемые действия.
- Допускается использование требуемых подпрограмм без параметров и локальных переменных.
- В отчете должно быть представлено полное тестовое покрытие. Приведены результаты тестовых прогонов. Например, с использованием скриншотов.
- При вводе некорректного значения размера массива программа должна завершать работу с выдачей соответствующего сообщения, а не требовать повторного ввода данных до получения ввода корректного числа.

При невыполнении хотя бы одного из требований оценка снижается.

6–7 баллов

При разработке программы на данную оценку необходимо учитывать все требования, предъявляемые на предшествующие оценки.

- В программе необходимо использовать подпрограммы с передачей аргументов через параметры, отображаемые на стек.

- Внутри подпрограмм необходимо использовать локальные переменные, которые при компиляции отображаются на стек.
- В местах вызова функции добавить комментарии, описывающие передачу фактических параметров и перенос возвращаемого результата. При этом необходимо отметить, какая переменная или результат какого выражения соответствует тому или иному фактическому параметру.
- Информацию о проведенных изменениях отобразить в отчете наряду с информацией, необходимой на предыдущую оценку.

При невыполнении хотя бы одного из требований оценка снижается на балл.

8 баллов

При разработке программы на данную оценку необходимо учитывать все требования, предъявляемые на предшествующие оценки.

- Разработанные подпрограммы должны поддерживать многократное использование с различными наборами исходных данных, включая возможность подключения различных исходных и результирующих массивов. То есть, поддерживать работу с формальными и фактическими параметрами.
- Реализовать автоматизированное тестирование за счет создания **дополнительной отдельной тестовой программы**, осуществляющей прогон подпрограммы обработки массивов с различными тестовыми данными (вместо ввода данных). Осуществить прогон тестов обеспечивающих покрытие различных ситуаций. Тестовые данные можно формировать в различных исходных массивах.
- Программа должна быть разбита на несколько единиц компиляции (ассемблерных файлов). При этом подпрограммы ввода-вывода должны составлять унифицированные модули, используемые повторно как в программе, осуществляющей ввод или генерацию исходных данных, так и в программе, осуществляющей тестовое покрытие.
- Добавить информацию о проведенных изменениях в отчет.

При невыполнении хотя бы одного из требований оценка снижается.

9 баллов

При разработке программы на данную оценку необходимо учитывать все требования, предъявляемые на предшествующие оценки.

- 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 массивами и другими необходимыми параметрами.
- Допускается реализация в виде собственных макросов и других фрагментов программы.
- Макросы должны быть выделены в отдельную автономную библиотеку

При невыполнении хотя бы одного из требований оценка снижается.

10 баллов

При разработке программы на данную оценку необходимо учитывать все требования, предъявляемые на предшествующие оценки.

- Вместо ввода данных с клавиатуры использовать ввод размера массива и его элементов из файла. Имя файла с исходными данными вводить с использованием клавиатуры.
- Расширить отчет, дополнив его новыми данными.

При невыполнении хотя бы одного из требований оценка снижается.