

Факультет компьютерных наук
Кафедра информационной безопасности

КОНТРОЛЬНЫЕ ЗАДАНИЯ ПО АССЕМБЛЕРУ

Задание 1.

Реализуйте вычисление заданного выражения как ассемблерную вставку в программе на языке высшего уровня (Паскаль, С++ и т.д.). Программа на языке высшего уровня должна запрашивать значения переменных и выводить результат на экран. Все переменные являются целочисленными.

- 1-1. $((A * X^2) \text{ div } (B * X^2)) \text{ div } (X * Y + 1)$
- 1-2. $A * X^5 + B * X^3 + C * X + D \text{ div } X$
- 1-3. $(A * X^4 + B * X^3 + C * X^2 + D * X) \text{ div } E$
- 1-4. $A * X^3 + B * X^2 + C \text{ div } X + (D \text{ div } X) \text{ div } X$
- 1-5. $(A * X^2 + B * X + C) \text{ div } (D * X + E)$
- 1-6. $(A * X + B) \text{ div } (C * X^2 + D * X + E)$
- 1-7. $((A * X + B) \text{ div } (C * X + D)) \text{ div } E$
- 1-8. $((A * X \text{ div } B) * X \text{ div } C) * X \text{ div } D) * X$
- 1-9. $A * X^2 + B * X * Y + C * Y^2 + D * X + E * Y + A$
- 1-10. $A * X * Y * Z + B * X * Y + C * Y * Z + D * X * Z + A$
- 1-11. $((A * X) \text{ div } (B * X * Y)) \text{ div } (X + 1)$
- 1-12. $A * (X^5 + B * X^3) + C * X + D \text{ div } X^2$
- 1-13. $A * X^4 + B * X^3 + C * X^2 + (D * X) \text{ div } E$
- 1-14. $(A * X^3 + B * X^2 + C \text{ div } X + D \text{ div } X) \text{ div } X$
- 1-15. $(A * X^2 + B * X) * Y + C \text{ div } (D * X + E)$
- 1-16. $(A * X + B) \text{ div } (C * X^2) + D * X + E$
- 1-17. $(A * X + B) \text{ div } ((C * X + D) \text{ div } E)$
- 1-18. $((A * X \text{ div } B) + X \text{ div } C) * X \text{ div } D) * X$
- 1-19. $(A * X^2 + B * X * Y + C) * Y^2 + D * X + E * Y + A$
- 1-20. $A * (X * Y * Z + B * X * Y + C * Y * Z) \text{ div } (D * X * Z) + A$

Задание 2.

Реализуйте вычисление заданного выражения как ассемблерную вставку в программе на языке высшего уровня (Паскаль, С++ и т.д.). Программа на языке высшего уровня должна запрашивать значения переменных и выводить результат на экран. Все переменные (X, Y) являются целочисленными. A, B, C – массивы. Если у массива указан один индекс, значит он одномерный, если два – двумерный. Индексы $i, j = 1..3$.

- 2-1. $\sum(A_i * X^2) + \sum(B_i * Y)$
- 2-2. $\sum(A_i * X) + \sum(B_i * X^2)$
- 2-3. $\sum(A_i * X) + \sum(B_i * XY) + (\sum C_i) * Y$
- 2-4. $\sum(A_i * B_i) * X + \sum(C_i) * Y$
- 2-5. $\sum(|A_i| * |B_i|) * X - \sum(|C_i|) * Y$
- 2-6. $\sum(|A_i|) * X + \sum(B_i) * Y$
- 2-7. $\sum(|A_i|) * X + \sum(|B_i|) * X * Y + \sum(C_i)$
- 2-8. $(\sum(A_i) + \sum(B_i)) * X * Y$
- 2-9. $|\sum(A_i + B_i)| * X + X^2$
- 2-10. $|\sum(A_i) * X + \sum(B_i) * Y|$
- 2-11. $\sum(A_{ij} * X) + \sum(B_i * Y)$
- 2-12. $\sum(A_i * X) + \sum(B_{ij} * X^2)$
- 2-13. $\sum(A_{ij} * X) + \sum(B_i * XY) + (\sum C_i) * Y$
- 2-14. $\sum(A_i * B_{ij}) * X + \sum(C_i) * Y$
- 2-15. $\sum(|A_i| * |B_{ij}|) * X - \sum(|C_i|) * Y$
- 2-16. $\sum(|A_{ij}|) * X + \sum(B_i) * Y$
- 2-17. $\sum(|A_i|) * X + \sum(|B_i|) * X * Y + \sum(C_{ij})$
- 2-18. $(\sum(A_i) + \sum(B_{ij})) * X * Y$
- 2-19. $|\sum(A_i + B_{ij})| * X + X^2$
- 2-20. $|\sum(A_i) * X + \sum(B_{ij}) * Y|$

Задание 3.

Программы реализуются как ассемблерные вставки в языке высшего уровня (Паскаль, С++ и т.д.).

- 3-1 С помощью логических операций реализуйте запись битов в данной ячейки памяти, размером в 2 байта, в обратном порядке.
- 3-2 Реализуйте логическую арифметику, при которой в одном байте хранилось бы 8 логических переменных.
- 3-3 Реализуйте логическую арифметику, при которой в одном слове хранилось бы 16 логических переменных.
- 3-4 Реализуйте с помощью логических операций вставку заданного значения (0 или 1) в любой бит байта со сдвигом вправо и потерей крайнего бита.
- 3-5 Реализуйте с помощью логических операций вставку заданного значения (0 или 1) в любой бит байта со сдвигом влево и потерей крайнего бита.
- 3-6 Реализуйте с помощью логических операций вставку заданного значения (0 или 1) в любой бита байта с потерей исходного значения.
- 3-7 Реализуйте с помощью логических симметричное отображение младшего байта слова на старший байт с потерей исходных значений.
- 3-8 Реализуйте с помощью логических симметричное отображение старшего байта слова на младший байт с потерей исходных значений.
- 3-9 Реализуйте с помощью логических операцию, при которой старший байт слова логически складывается с младшим байтом, записанным в обратном порядке.
- 3-10 Реализуйте с помощью логических операцию, при которой старший байт слова логически умножается на младший байт, записанный в обратном порядке.
- 3-11 Реализуйте с помощью логических операцию, при которой младший байт слова логически складывается со старшим байтом записанным в обратном порядке.
- 3-12 Реализуйте с помощью логических операцию, при которой младший байт слова логически умножается на младшим байтом, записанный в обратном порядке.
- 3-13 Одномерный вектор имеет восемь координат, каждая из которых может принимать значения 0 или 1. Реализуйте с помощью битовых операций векторную алгебру (сложение, вычитание и скалярное произведение векторов).
- 3-14 С помощью битовых операций реализуйте перестановку двух любых битов в данном слове местами. Остальные биты остаются неизменными.
- 3-15 Заданы два числа ≤ 15 . Найти их сумму и с помощью битовых операций записать оба слагаемых и их сумму одно за другим в ячейку памяти размером в два байта без пропуска битов.
- 3-16 Заданы два числа ≤ 15 . Найти их произведение и с помощью битовых операций записать оба множителя и их произведение одно за другим в ячейку памяти размером в два байта без пропуска битов.
- 3-17 Заданы два числа ≤ 255 . Найти целую часть от деления первого числа на второе и остаток деления. Записать две полученные величины одно за другим без пропуска битов в ячейку памяти размером в два байта.
- 3-18 Дано слово из трех букв (например ФКН). С помощью битовых операций запишите последовательность ASCII кодов его букв в ячейку памяти минимального размера. Слово может быть и другим, программа должна его запрашивать. Запрашивать можно на языке высокого уровня.
- 3-19 Дано пятизначное число. (Может быть например такое 00001). С помощью битовых операций запишите его в ячейку памяти минимального размера. Запрашивать число можно на языке высокого уровня.
- 3-20 В двухбайтной ячейке памяти записана последовательность нулей и единиц. С помощью битовых операций переставьте биты так чтобы сначала шли все единицы, а затем все нули.

Вопросы к зачету по ассемблеру

1. Место языков ассемблера среди языков программирования.
2. Структура МП Intel 80x86: используемые регистры.
3. Структура МП Intel 80x86: операционное устройство и шинный интерфейс.
4. Размещение данных в памяти. Сегментация памяти.
5. Структура регистра флагов. Команды установки флагов.
6. Структура и форматы команд МП Intel 80x86. Команды пересылки данных.
7. Способы адресации в командах МП Intel 80x86.
8. Представление данных в IBM PC: целые числа.
9. Представление данных в IBM PC: двоично-десятичные числа.
10. Представление данных в IBM PC: алфавитно-цифровые данные.
11. Представление данных в IBM PC: вещественные данные.
12. Система команд МП: команды сложения и вычитания.
13. Команды умножения и деления чисел с ФТ.
14. Структура команд МП: базовая, индексная и косвенная адресации.
15. Логические команды обработки битов.
16. Команды сдвигов и их использование.

17. Команды передачи управления: безусловные переходы. Адресация в переходах.
18. Команды передачи управления: условные переходы.
19. Команды передачи управления: организация циклов.
20. Стек. Команды работы со стеком.
21. Элементарные конструкции языка ассемблера: алфавит, ключевые слова.
22. Элементарные конструкции языка ассемблера: числа, символьные данные.
23. Элементарные конструкции языка ассемблера: имена, метки.
24. Элементарные конструкции языка ассемблера: выражения и их использование.
25. Предложения языка ассемблера: комментарии.
26. Предложения языка ассемблера: команды.
27. Предложения языка ассемблера: директивы.
28. Структура файла ассемблер-программы. Директивы оформления программы.
29. Структура файла ассемблер-программы: односегментные и многосегментные файлы.
30. Использование прерываний в ассемблер-программах.
31. Операторы в командах языка ассемблера.
32. Блочная структура программы: правила описания и вызова процедур.
33. Блочная структура программы: расположение процедур в исходном файле.
34. Блочная структура программы: внутренние и внешние процедуры.
35. Способы передачи параметров между процедурой и вызывающей программой.
36. Передача параметров между процедурой и вызывающей программой. Проблема сохранения регистров.
37. Программные пакеты MASM и TASM: этапы обработки задания (подготовка исходного файла и его трансляция).
38. Программные пакеты MASM и TASM: этапы обработки задания (компоновка объектного модуля и отладка программы).
39. Программные пакеты MASM и TASM: общие функции и различия.
40. Модели памяти и их использование в TASM.
41. Макросы: макроопределения и их использование.
42. Макросы: использование параметров и комментарии.

СПИСОК ЛИТЕРАТУРЫ

Юров В.А. Ассемблер. Учебник для вузов. СПб.: изд-во Питер, 2003

Магда Ю. Ассемблер для процессоров Intel Pentium. Учебное пособие. СПб: изд-во Питер, 2006

Богаченко Н.Ф., Лавров Д.Н., Ракицкий Ю.С. Ассемблер в примерах и задачах. Учебно-методическое пособие. Часть 1.: Омск, изд-во Ом. гос. ун-та, 2011