

задачи с экзаменов и Бородаченковой

5 команд значит 5 инструкций число строк макроса неважно, что такое макрорасширение найдёте в файлах ответы к экзаменам искать там где экзамены писать на листочке
посмотреть файл Задачи 10_шпора

10. Описать макрос `ONES v`, который записывает в регистр `AL` количество разрядов `v`, равных "1". Здесь `v` – имя переменной типа `byte`, `word` или `dword`. Можно использовать любые регистры. Содержимое `v` не портить. Ошибки в параметре не проверять. Макрорасширение должно содержать не более 5 команд. (Привести ответ на обратной стороне листа).

10. Описать макрос `ZEROS v`, который записывает в `BL` количество разрядов `v`, равных "0". Здесь `v` – имя переменной типа `byte`, `word` или `dword`. Можно использовать любые регистры. Содержимое `v` не портить. Ошибки в параметре не проверять. Макрорасширение должно содержать не более 5 команд. (Привести ответ на обратной стороне листа).

10. Описать макроопределение `SUM X`, которое записывает в регистр `EAX` сумму элементов массива `X` (массив описан с помощью конструкции `dup`). Элементы массива – байты, числа со знаком. Регистры (кроме `EAX`) не портить. Считать, что сумма уместится в двойное слово. Наличие и правильность передачи параметра не проверять. Описать массив `B` из 50 элементов и выписать макрорасширение для вызова `SUM B`. (Привести ответ на обратной стороне листа.)

8. Описать макрос `Print1 X`. Здесь `X` – это список $\langle X_1, X_2, \dots, X_k \rangle$, где X_i – имена переменных размером в байт, слово или двойное слово. Макрос печатает значения всех байтовых переменных из списка `X` как числа со знаком по макрокоманде `OUTI`. Считать, что `OUTI` допускает только операнд в формате `dword`. Регистры не портить. Правильность и наличие параметра `X` не проверять.

экзамен давний некоторые ещё времён 16 битного ассемблера, подсказка использовать `@substr(строка, начало[, длина])` строка начинается с 1 если задач недостаточно то таких номеров в сборнике найти можно много

8. Написать макроопределение с заголовком.

Maximum macro X,Y

✓ Это макроопределение должно присваивать параметру X наибольшую из *знаковых* величин X и Y. Параметры X и Y должны быть только переменными одинакового размера в байт или слово. Макроопределение обязательно должно проверять наличие и допустимость своих параметров, выдавая необходимые диагностики об ошибках. (Ответ на обороте).

↓ 7. Описать макроопределение с заголовком JDIF macro LN, M (где LN – параметр-список $\langle N_1, N_2, \dots, N_k \rangle$ явно заданных знаковых чисел, а M – метка). Макроопределение делает переход на метку M, если значение регистра AX отлично от всех *отрицательных* чисел из параметра-списка LN или если в LN нет отрицательных чисел. Макроопределение не проверяет наличие и правильность типов своих параметров, не сохраняет используемые регистры. (Ответ на обороте.)

7. Привести макроопределение с именем Bytes, обращение к которому имеет вид, показанный ниже слева (здесь в параметр-списке: $k \geq 1$, x_i – имя, v_i – знак вопроса ? или число, между x_i и v_i стоит пробел). Это макроопределение должно формировать макрорасширение, показанное ниже справа. Макроопределение *не должно* проверять наличие и правильность типов своих параметров. (Ответ на обороте).

Bytes $\langle x_1 \ v_1 \dots x_k \ v_k \rangle \rightarrow \begin{cases} x_1 \text{ db } v_1 \\ \dots \dots \dots \\ x_k \text{ db } v_k \end{cases}$