

Перелік запитань по курсу  
"Системне програмування 1. Програмування на Асемблері"

Одна контрольна робота містить **4 питання**. Кожне питання має 28 варіантів. Номер варіанту згідно списку у журналі.

**Перше питання. Варіанти:**

1. Що таке Асемблер? Для чого потрібен Асемблер?
2. Що обов'язково повинен знати програміст на Асемблері на відміну від програміста мови високого рівня? Які джерела інформації для програмістів на Асемблері можна вважати найважливішими?
3. Як розпочати проект на асемблері у середовищі MS Visual Studio?
4. Як налагоджувати програму на Асемблері?
5. Що таке дизасемблер?
6. Базове середовище виконання програм процесорів 32-бітової архітектури Intel
7. Базове середовище виконання програм процесорів 64-бітової архітектури Intel
8. Ресурси середовища IA-32 для виконання програм
9. Ресурси середовища Intel64 для виконання програм
10. Режими роботи процесорів архітектури x86
11. Який адресний простір архітектури IA-32 і чим він визначається?
12. Який адресний простір архітектури Intel64 і чим він визначається?
13. Фізична пам'ять. Як узнати яку розрядність адрес підтримує процесор?
14. Моделі пам'яті у процесорах архітектури x86
15. Сегментована пам'ять
16. Flat - модель пам'яті
17. Сторінкова пам'ять
18. Позиційні системи числення. Поняття та основні формули
19. Способи переведення у іншу систему числення. Приклади
20. Способи представлення від'ємних чисел. Приклади
21. Додатковий код. Приклади
22. Зміщений код. Приклади
23. Формати з плаваючою точкою. Математичний запис числа. Стандарти форматів
24. Стандартний двійковий 32-бітовий формат з плаваючою точкою
25. Стандартний двійковий 64-бітовий формат з плаваючою точкою
26. Стандартний двійковий 80-бітовий формат з плаваючою точкою
27. Порівняння основних стандартних двійкових форматів з плаваючою точкою
28. Середовище x87 FPU
29. Регістри і стек даних x87 FPU. Приклад програми

**Друге питання. Варіанти:**

1. Регістри процесора архітектури IA-32. Приклади їхнього використання
2. Регістри процесора архітектури Intel64
3. Регістри загального призначення. Приклади їхнього використання
4. Сегментні регістри

5. Регістр EFLAGS (прапорців)
6. Що таке регістр ESP і як він використовується?
7. Регістр EIP
8. Операнди команд та способи адресації операндів
9. Безпосередні операнди команд
10. Регістрові операнди команд
11. Способи вказування комірок пам'яті у якості операндів команд
12. Що таке опосередкована адресація стосовно вказування операнду команди?
13. Що таке база, індекс, зміщення, масштаб у операнді команди?
14. Стек: що це таке, як він працює і де використовується.
15. Структура вихідного тексту програми на Асемблері
16. Скелет тексту програми на Асемблері для Windows
17. Директиви Асемблера. Найпопулярніші на вашу думку директиви
18. Директиви вказування типу процесора та моделі пам'яті
19. Директиви створення даних
20. Директива INVOKE. Приклади
21. Директива ALIGN. Приклади
22. Директива LOCAL. Приклади
23. Директива EQU. Приклади
24. Базові типи даних архітектури процесорів x86
25. Числові типи даних
26. Упаковані SIMD типи даних
27. Типи даних SSE
28. Типи даних AVX

**Третє питання. Варіанти:**

1. Основні групи команд процесорів архітектури x86
2. Команда CPUID та її використання. Приклади
3. Команди копіювання даних
4. Команда PUSH. Як вона працює
5. Команда POP. Як вона працює
6. Команди додавання цілих чисел
7. Команди віднімання цілих чисел
8. Команди множення цілих чисел
9. Команди ділення цілих чисел
10. Побітові логічні команди
11. Програмування читання та запису окремих бітів
12. Команди зсувів
13. Команди кільцевих зсувів
14. Команди переходів
15. Програмування умовних операцій типу IF-THEN-ELSE на Асемблері
16. Програмування циклів на Асемблері
17. Програмування циклів з передумовою
18. Програмування циклів із постумовою

19. Програмування циклу з недоторканістю біту переносу CF
20. Програмування циклу на основі команд LOOP
21. Програмування вкладених циклів
22. Команди обробки рядків даних
23. Команди STOS, STOSx
24. Команди MOVS, MOVSx
25. Префікс повторення REP та його різновиди у командах обробки рядків даних
26. Команди MOVUPS та MOVAPS. Приклад використання
27. Команда SHUFPS та її застосування
28. Команди x87 FPU. Приклад програми

**Четверте питання. Варіанти:**

1. Що таке структурованість та модульність програм. Наведіть приклади
2. Макроси. Приклади програмування
3. Процедури. Визначення, виклик. Приклади програмування
4. Способи передавання значень параметрів процедурам
5. Локальні дані процедур. Приклади
6. Пролог та епілог процедури. Приклади
7. Стековий кадр процедури
8. Написання модульних програм на Асемблері
9. Можливості використання мов високого рівня сумісно з Асемблером
10. Конвенції виклику процедур
11. Конвенція **cdecl**. Приклад програмування
12. Конвенція **stdcall**. Приклад програмування
13. Асемблерні вставки у мовах високого рівня
14. Команди SIMD
15. Розширення MMX
16. Розширення SSE
17. Розширення AVX
18. Векторні команди SSE. Приклади
19. "Вертикальне" та "горизонтальне" додавання елементів векторів
20. Векторні команди множення SSE. Приклад використання
21. Векторні команди додавання SSE. Приклад використання
22. Команди AVX
23. Особливості команд AVX по відношенню до SSE
24. Еволюція SIMD-розширень системи команд процесорів архітектури x86
25. Програмування чисел підвищеної розрядності
26. Використання функцій API Windows. Приклади програмування
27. Програмування на Асемблері роботи з файлами
28. Використання глобальної динамічної пам'яті