

1. В чому полягають основні властивості і особливості позиційних систем числення?
2. Що називають основою системи числення?

$$2^n \quad \text{або} \quad -10 \leq n \leq 10$$

3. Записати число в нормальній формі.
4. Сформулювати правило перекладу цілих чисел з однієї системи числення в іншу методом розподілу на основу.
5. Сформулювати правило перекладу дробових чисел з однієї системи числення в іншу методом множення на основу.
6. Сформулювати правило перекладу чисел з двійкової системи у вісімкову.
7. Як виконати над числами наступні арифметичні операції: $A + B + C$; $A - B$; $C - D$; $E - F$; $B * E$; $C * D$; $F : E$; $A : E$. Операцію віднімання виконувати як складання алгебри, представляючи негативні числа в зворотному коді
8. Чим викликано застосування додаткового і зворотного кодів?
9. В чому відмінності представлення чисел в природній і нормальній формах? Яка форма і коли зручніше?
10. Як провести складання чисел. $A = 0.1100 * 10^{-101}$, $B = 0.1001 * 10^{-001}$
11. Що таке *алгоритм*?
12. Перерахувати базові структури алгоритмів.
13. Визначити, що являють собою ланцюг та *розгалуження*.
14. Що таке *цикл* як базова структура алгоритму?
15. Які існують різновиди циклів?
16. Що таке «базові типи даних»?
17. Які існують цілочисельні типи у мові C? Скільки байтів пам'яті займає тип *int*?
18. Назвати дійсні типи мови C? Скільки байтів пам'яті займає тип *float*?
19. Як представляються символічні дані у мові C?
20. Що являє собою тип *void*?
21. Що являє собою програма на мові C? Яка структура C-програми?
22. Що таке «операція» та що таке «операнд»? Яка операція називається унарною, а яка – бінарною?
23. Що являє собою вираз?
24. Які арифметичні операції використовуються у мові C?
25. Чи існує логічний тип у мові C? Які значення використовуються для представлення логічних значень?
26. Які використовують операції порівняння та які – логічні операції?
27. Які операції відносяться до операцій присвоювання?
28. Які групи операцій присвоювання існують у мові C? Навести приклади.
29. Чи є оператор програмною одиницею?
30. Які групи основних керуючих конструкцій являють собою оператори?
31. Що таке «пустий оператор» та для чого він використовується?
32. Які існують оператори простої послідовності?
33. Що таке *умовна конструкція*?
34. Структури умовного оператора.
35. Як працює умовний оператор?
36. Що являє собою структура оператора множинного вибору?
37. Як працює оператор множинного вибору?
38. Яка структура програми на мові C?
39. Навіщо потрібна директива *#include*?
40. Що таке *main()*?
41. Перерахувати скалярні типи даних мови C.
42. Що визначає тип даних?
43. Що таке явне і неявне приведення типів? Як і коли воно використовується?
44. Що таке константа? Знайдіть константи у наведених програмах.
45. Що таке змінна?

46. Як проініціалізувати змінну?
47. Чим відрізняється оператор від операції?
48. Чим відрізняються унарні операції від бінарних?
49. Які операції відносяться до арифметичних? Який пріоритет кожної з них?
50. Який порядок виконання операцій у випадку їх однакового пріоритету?
51. Як виконується операція ділення у випадку цілочисельних операндів та у випадку, коли хоч би один з операндів дійсний?
52. Що таке вираз?
53. Яке значення обчислює операція присвоювання?
54. В якому порядку виконується присвоювання у випадку, якщо у виразі їх декілька?
55. Як і навіщо використовуються додаткові операції присвоювання?
56. Чим відрізняються префіксна форма операції інкремента або декремента від постфіксної?
57. Які функції використовуються для вводу інформації? Назвіть їх відмінні особливості.
58. Які функції використовуються для виводу інформації? Назвіть їх відмінні особливості.
59. Чому функції `scanf()` та `printf()` називаються функціями форматного вводу та виводу? Як вони працюють?
60. Чим відрізняється керуюча строка функції `scanf()` від керуючої строки функції `printf()`?
61. Що таке специфікатор формату? Навіщо він потрібен?
62. Які параметри вказуються функцією `scanf()` після керуючої строки? Скільки їх повинно бути?
63. Які наслідки невідповідності типу зчитуваної функцією `scanf()` змінної специфікатору типу?
64. Які параметри вказуються функції `printf()` після керуючої строки? Скільки їх повинно бути?
65. Які наслідки невідповідності типу виводимого функцією `printf()` значення специфікатору типу?
66. Що таке керуючі символи? Навіщо вони потрібні? Наведіть приклади.
67. Що таке *цикл*?
68. Які існують два типи циклів?
69. Структура оператору циклу з параметром.
70. Як працює цикл з параметром?
71. Вказати особливості оператора циклу с параметром у мові C?
72. Які цикли відносяться до ітераційних?
73. Навести структуру оператора циклу з передумовою.
74. Як працює цикл з передумовою?
75. Яку структуру має оператор циклу з постумовою?
76. Як працює цикл з постумовою?
77. Чим відрізняється умовна операція від умовного оператору?
78. Що таке повна і неповна форма умовного оператору?
79. Чи може існувати неповна форма умовної операції?
80. Чи потрібно писати "else", якщо при виконанні умови виконується оператор `return`?
81. Вирази якого типу можуть визначати умови?
82. Які значення виразів, що визначають умови, вважаються істинними, а які хибними?
83. Які операції відносяться до операцій відношення?
84. Чим відрізняються операції `"=="` від операції `"="`?
85. Які операції відносяться до логічних? Який пріоритет їх виконання?
86. Якою операцією можна замінити операцію `"&&"` ?
87. Якою операцією можна замінити операцію `"||"` ?
88. Чому може дорівнювати значення виразу відношення або логічного виразу?
89. Як правильно порівняти на рівність дійсні числа?
90. Як правильно перевірити входження значення у певний діапазон?
91. Як перевірити певне цілочисельне значення на рівність нулю?
92. Як перевірити відмінність цілочисельного значення від нуля?
93. Коли використовується вложення умовних операторів?
94. Як правильно записати вложені умовні оператори?
95. Що являє собою оператор `switch`? Як їм користуватися?
96. Як записати оператор `switch` з допомогою умовних операторів?
97. Які види циклів снують? Які оператори циклу існують у мові C?

98. Чим відрізняються цикл з передумовою від циклу з постумовою?
99. Коли необхідно використовувати цикл з передумовою, а коли з постумовою? Наведіть приклади.
100. Які цикли з передумовою існують у мові C?
101. Скільки операторів містять у собі тіло циклу у мові C?
102. Як правильно записати цикл з постумовою на мові C?
103. Як задати нескінченний цикл? Навіщо він потрібен? Як з нього вийти?
104. Яким повинно бути значення виразу, який визначає умову виконання циклу, для завершення циклу?
105. Яким повинно бути значення виразу, який визначає умову виконання циклу, для виконання тіла циклу?
106. До чого призведе неправильне завдання виразу, який визначає умови виконання циклу?
107. Чи може бути відсутнім тіло циклу? Якщо може, то наведіть приклади таких циклів.
108. Чим відрізняється оператор *while* від оператора *if*?
109. Який порядок дій при виконанні циклу *for*?
110. Як організувати арифметичний цикл з допомогою циклу *for*?
111. Запишіть алгоритм, який визначається циклом *for*, з допомогою циклу *while*.
112. Що таке вкладений цикл?
113. Скільки разів у загальній складності виконується тіло вкладеного циклу?
114. Як і коли використовуються оператори *break* то *continue*?
115. Що таке рекурентні обчислення? Коли вони використовуються? Як їх програмувати?
116. Що таке функція?
117. Що таке функція типу *void*?
118. Що таке прототип функції?
119. Чим відрізняється прототип функції від виклику функції?
120. Чим відрізняється описання функції від визначення функції?
121. Критерії відповідності формальних і фактичних параметрів.
122. Відповідність типів формальних і фактичних параметрів.
123. Навіщо потрібен оператор *return*?
124. Чи може у функції бути декілька операторів *return*?
125. Коли необхідно писати оператор *return* у функції типу *void*?
126. Що таке побочний ефект функції?
127. Коли використовують формальні параметри-вказівники? Якими у цьому випадку повинні бути фактичні параметри?
128. Як передати масив у функцію? Як передати у функцію матрицю?
129. Як описати функцію, яка дозволяє працювати одночасно з одновимірними масивами, і з матрицями?
130. Що таке параметр-константа? Коли використовують такі параметри
131. Що таке вказівник на функцію? Як його описати?
132. Як передати функцію у функцію? Що буде формальним параметром, а що фактичним? Як
133. звернутися до функції, яку передано через параметри?
134. Що таке рекурсія? Коли вона використовується? Що таке глибина рекурсії?
135. Як описати функцію зі змінною кількістю параметрів? Як використовувати таку функцію?
136. Які параметри можуть бути у функції *main()*? Як налагоджувати і тестувати програму, яка містить функцію *main()* з параметрами?