

**ТЕСТОВІ ЗАВДАННЯ**  
**З ДИСЦИПЛІНИ «АЛГОРИТМИ ТА СТРУКТУРИ ДАНИХ»**

| Текст завдання  |
|---|
| Алгоритм це є   |
| Алгоритм називається лінійним, якщо   |
| Розгалужений алгоритм – це алгоритм, який   |
| Алгоритм називається циклічним, якщо  |
| Алгоритми за формою подання можна поділити на: (вказіть невірний варіант)   |
| Алгоритм записаний у вигляді послідовності команд, що виражені у формі придатній для зчитування комп'ютером, які приводять його у дію для досягнення певного результату називають |
| Який з варіантів вірно відображає значення, яке зберігається в змінної var, якщо змінна була ініціалізована наступним чином (мова C):<br>signed short var = -13;                  |
| Який з варіантів вірно відображає значення, яке зберігається в змінної var, якщо змінна була ініціалізована наступним чином (мова C):<br>signed short var = -103;                 |
| Який з варіантів вірно відображає значення, яке зберігається в змінної var, якщо змінна була ініціалізована наступним чином (мова C):<br>signed char var = -13;                   |
| Який з варіантів вірно відображає значення, яке зберігається в змінної var, якщо змінна була ініціалізована наступним чином (мова C):<br>signed char var = -103;                  |
| Маємо код (мова C):<br>signed short var = 300;<br>signed char res = var;<br>Яке значення матиме res?  |

|   |
|---|
| <p>Маємо код (мова C):<br/> signed short var = 300;<br/> unsigned char res = var;<br/> Яке значення матиме res?</p>                 |
| <p>Маємо код (мова C):<br/> signed short var = 400;<br/> signed char res = var;<br/> Яке значення матиме res?</p>                   |
| <p>Маємо код (мова C):<br/> signed short var = 400;<br/> unsigned char res = var;<br/> Яке значення матиме res?</p>                 |
| <p>Маємо код (мова C):<br/> signed short var = 421;<br/> signed short res = (signed char)var;<br/> Яке значення матиме res?</p>     |
| <p>Маємо код (мова C):<br/> signed short var = 421;<br/> signed short res = (unsigned char)var;<br/> Яке значення матиме res?</p>   |
| <p>Маємо код (мова C):<br/> signed short var = 421;<br/> unsigned short res = (signed char)var;<br/> Яке значення матиме res?</p>   |
| <p>Маємо код (мова C):<br/> signed short var = 421;<br/> unsigned short res = (unsigned char)var;<br/> Яке значення матиме res?</p> |
| <p>Якого розміру змінна типу char у мові C?</p>   |
| <p>Якого розміру змінна типу signed char у мові C?</p>  |
| <p>Якого розміру змінна типу unsigned char у мові C?</p>  |
| <p>Якого розміру змінна типу short у мові C?</p>  |
| <p>Якого розміру змінна типу signed short у мові C?</p>   |
| <p>Якого розміру змінна типу unsigned short у мові C?</p>   |
| <p>Якого розміру змінна типу long у мові C?</p>   |
| <p>Якого розміру змінна типу signed long у мові C?</p>  |
| <p>Якого розміру змінна типу unsigned long у мові C?</p>  |
| <p>Якого розміру змінна типу long long у мові C?</p>  |

|   |
|---|
| C?  |
| Якого розміру змінна типу signed long long у мові C?  |
| Якого розміру змінна типу unsigned long long у мові C?  |
| Якого розміру змінна типу float у мові C?   |
| Якого розміру змінна типу double у мові C?  |
| Якого розміру змінна типу long double у мові C?   |
| Який тип даних займає 1 байт пам'яті (мова C)?  |
| Який тип даних займає 2 байти пам'яті (мова C)?   |
| Який тип даних займає 4 байти пам'яті (мова C)?   |
| Який тип даних займає 4 байти пам'яті (мова C)?   |
| Який тип даних займає 8 байтів пам'яті (мова C)?  |
| Який тип даних займає 10 байтів пам'яті (мова C)?   |
| Якого розміру змінна v (мова C)?<br><pre>struct var {     unsigned short v1:5;     unsigned short v2:7; } v;</pre>                          |
| Якого розміру змінна v (мова C)?<br><pre>struct var {     unsigned short v1:5;     unsigned short v2:7;     unsigned short v3:6; } v;</pre> |
| Якого розміру змінна v (мова C)?<br><pre>struct var {     unsigned long v1:5;     unsigned long v2:7;     unsigned long v3:6; } v;</pre>    |
| Якого розміру змінна v (мова C)?<br><pre>struct var {     unsigned char v1:5;     unsigned char v2:7;     unsigned char v3:6; } v;</pre>    |

Якого розміру змінна v (мова C)?

```
struct var {  
    unsigned long v1:5;  
    unsigned long v2:7;  
    unsigned long v3:6;  
    unsigned long v4:6;  
} v;
```

Якого розміру змінна v (мова C)?

```
struct var {  
    unsigned char v1:5;  
    unsigned char v2:7;  
    unsigned char v3:6;  
    unsigned char v4:6;  
} v;
```

Якого розміру змінна v (мова C)?

```
struct var {  
    unsigned char v1:2;  
    unsigned char v2:3;  
    unsigned char v3:1;  
} v;
```

Маємо код (мова C):

```
struct var {  
    unsigned short v1:4;  
    unsigned short v2:7;  
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;  
*ptr = 5;
```

Яке значення матиме v.v1?

Маємо код (мова C):

```
struct var {  
    unsigned short v1:4;  
    unsigned short v2:7;  
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;  
*ptr = 5;
```

Яке значення матиме v.v2?

Маємо код (мова C):

```
struct var {  
    unsigned short v1:4;
```

```
    unsigned short v2:7;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = 5*2048;
```

Яке значення матиме v.v1?

Маємо код (мова C):

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;
*ptr = 5*2048;
```

Яке значення матиме v.v2?

Маємо код (мова C):

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;
*ptr = -5;
```

Яке значення матиме v.v1?

Маємо код (мова C):

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;
*ptr = -5;
```

Яке значення матиме v.v2?

Маємо код (мова C):

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
```

```
unsigned short v3:4;
unsigned short v4:1;
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;
*ptr = -16;
```

Яке значення матиме v.v4 та v.v1?

Маємо код (мова C):

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;
*ptr = -16;
```

Яке значення матиме v.v4 та v.v2?

Маємо код (мова C):

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;
*ptr = -16;
```

Яке значення матиме v.v2 та v.v1?

Маємо код (мова C):

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v;
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;  
*ptr = -16;
```

Яке значення матиме v.v3 та v.v1?

Маємо код (мова C):

```
struct var {  
    unsigned short v1:4;  
    unsigned short v2:7;  
    unsigned short v3:4;  
    unsigned short v4:1;  
} v = {4, 3, 2, 1};
```

```
unsigned short *ptr;
```

```
ptr = (unsigned short *)&v;
```

Яке значення буде зберігати комірка на яку вказує ptr?

Маємо код (мова C):

```
struct var {  
    unsigned short v1:4;  
    unsigned short v2:7;  
    unsigned short v3:4;  
    unsigned short v4:1;  
} v = {4, 3, 2, 1};
```

```
signed short *ptr;
```

```
ptr = (signed short *)&v;
```

Яке значення буде зберігати комірка на яку вказує ptr?

Маємо код (мова C):

```
struct var {  
    unsigned short v1:4;  
    unsigned short v2:7;  
    unsigned short v3:4;  
    unsigned short v4:1;  
} v = {4, 3, 2, 1};
```

```
unsigned short *ptr;
```

```
ptr = (signed short *)&v;
```

Яке значення буде зберігати комірка на яку вказує ptr?

Маємо код (мова C):

```
struct var {  
    unsigned short v1:4;
```

```
unsigned short v2:7;
unsigned short v3:4;
unsigned short v4:1;
} v = {4, 3, 2, 1};
```

```
signed short *ptr;
```

```
ptr = (unsigned short *)&v;
```

Яке значення буде зберігати комірка на яку вказує ptr?

Маємо код (мова C):

```
struct var {
    unsigned char v1:1;
    unsigned char v2:1;
    unsigned char v3:1;
    unsigned char v4:1;
    unsigned char v5:1;
    unsigned char v6:1;
    unsigned char v7:1;
    unsigned char v8:1;
} v;
```

```
signed char *ptr;
```

```
ptr = (signed char *)&v;
```

```
*ptr = -5;
```

Яке поле структури необхідно перевірити, щоб визначити знак заданого числа?

Маємо код (мова C):

```
struct var {
    unsigned char v1:1;
    unsigned char v2:1;
    unsigned char v3:1;
    unsigned char v4:1;
    unsigned char v5:1;
    unsigned char v6:1;
    unsigned char v7:1;
    unsigned char v8:1;
} v;
```

```
signed char *ptr;
```

```
ptr = (signed char *)&v;
```

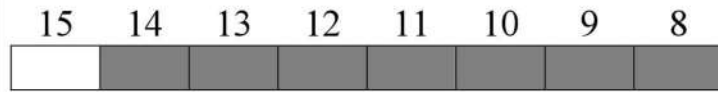
```
*ptr = 23;
```


Яке поле структури необхідно перевірити,



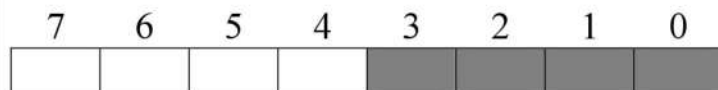
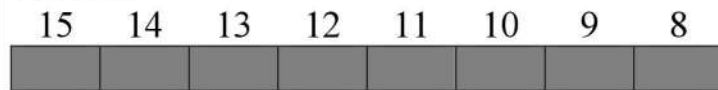
щоб визначити парність числа?


Яка структура на мові С відповідає  
представленій структурі розподілу бітів в  
пам'яті?



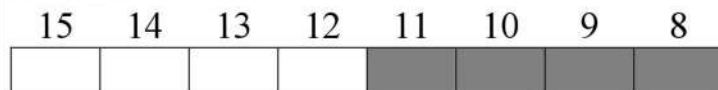
 - зайняті комірки пам'яті


Яка структура на мові С відповідає  
представленій структурі розподілу бітів в  
пам'яті?



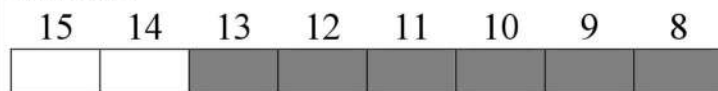
 - зайняті комірки пам'яті


Яка структура на мові С відповідає  
представленій структурі розподілу бітів в  
пам'яті?



 - зайняті комірки пам'яті

Яка структура на мові С відповідає  
представленій структурі розподілу бітів в  
пам'яті?




 - зайняті комірки пам'яті

Яка структура на мові С відповідає  
представленій структурі розподілу бітів в  
пам'яті?

|    |    |    |    |    |    |   |   |
|----|----|----|----|----|----|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|    |    |    |    |    |    |   |   |


|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|   |   |   |   |   |   |   |   |

 - зайняті комірки пам'яті

Яка структура на мові С відповідає  
представленій структурі розподілу бітів в  
пам'яті?

|    |    |    |    |    |    |   |   |
|----|----|----|----|----|----|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|    |    |    |    |    |    |   |   |

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|   |   |   |   |   |   |   |   |

 - зайняті комірки пам'яті

Яка змінна на мові С відповідає  
представленій структурі розподілу бітів в  
пам'яті?

|      |    |         |    |    |    |    |    |
|------|----|---------|----|----|----|----|----|
| знак |    | мантиса |    |    |    |    |    |
| 31   | 30 | 29      | 28 | 27 | 26 | 25 | 24 |

|         |    |    |    |    |    |    |    |
|---------|----|----|----|----|----|----|----|
| порядок |    |    |    |    |    |    |    |
| 23      | 22 | 21 | 20 | 19 | 18 | 17 | 16 |

|         |    |    |    |    |    |   |   |
|---------|----|----|----|----|----|---|---|
| порядок |    |    |    |    |    |   |   |
| 15      | 14 | 13 | 12 | 11 | 10 | 9 | 8 |

|         |   |   |   |   |   |   |   |
|---------|---|---|---|---|---|---|---|
| порядок |   |   |   |   |   |   |   |
| 7       | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Яка змінна на мові С відповідає  
представленій структурі розподілу бітів в  
пам'яті?

|      |    |          |    |    |    |    |    |
|------|----|----------|----|----|----|----|----|
| знак |    | значення |    |    |    |    |    |
| 31   | 30 | 29       | 28 | 27 | 26 | 25 | 24 |

|          |    |    |    |    |    |    |    |
|----------|----|----|----|----|----|----|----|
| значення |    |    |    |    |    |    |    |
| 23       | 22 | 21 | 20 | 19 | 18 | 17 | 16 |

|          |    |    |    |    |    |   |   |
|----------|----|----|----|----|----|---|---|
| значення |    |    |    |    |    |   |   |
| 15       | 14 | 13 | 12 | 11 | 10 | 9 | 8 |

значення

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = 5;
printf("%d\n", v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = 5;
printf("%d\n", v.v2);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = 5*2048;
printf("%d\n", v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = 5*2048;
printf("%d\n", v.v2);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = -5;
printf("%d\n", v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = -5;
printf("%d\n", v.v2);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v;
```

```
unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = -16;
printf("%d, %d\n", v.v4, v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = -16;
printf("%d, %d\n", v.v4, v.v2);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v;

unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = -16;
printf("%d, %d\n", v.v2, v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v;
```

```
unsigned short *ptr;

ptr = (unsigned short *)&v;
*ptr = -16;
printf("%d, %d\n", v.v3, v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v = {4, 3, 2, 1};

unsigned short *ptr;

ptr = (unsigned short *)&v;
printf("%d\n", *ptr);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v = {4, 3, 2, 1};

signed short *ptr;

ptr = (signed short *)&v;
printf("%d\n", *ptr);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {
    unsigned short v1:4;
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v = {4, 3, 2, 1};

unsigned short *ptr;
```

```
ptr = (signed short *)&v;  
printf("%d\n", *ptr);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
struct var {  
    unsigned short v1:4;  
    unsigned short v2:7;  
    unsigned short v3:4;  
    unsigned short v4:1;  
} v = {4, 3, 2, 1};  
  
signed short *ptr;  
  
ptr = (unsigned short *)&v;  
printf("%d\n", *ptr);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {  
    struct {  
        unsigned short v1:4;  
        unsigned short v2:7;  
    } v;  
    unsigned short p;  
} u;  
  
u.p = 5;  
printf("%d\n", u.v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {  
    struct {  
        unsigned short v1:4;  
        unsigned short v2:7;  
    } v;  
    unsigned short p;  
} u;  
  
u.p = 5;  
printf("%d\n", u.v.v2);
```

Що буде виведено на екран при виконанні

фрагмента коду:

```
union {
    struct {
        unsigned short v1:4;
        unsigned short v2:7;
    } v;
    unsigned short p;
} u;

u.p = 5*2048;
printf("%d\n", u.v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {
    struct {
        unsigned short v1:4;
        unsigned short v2:7;
    } v;
    unsigned short p;
} u;

u.p = 5*2048;
printf("%d\n", u.v.v2);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {
    struct var {
        unsigned short v1:4;
        unsigned short v2:7;
    } v;
    unsigned short p;
} u;

u.p = -5;
printf("%d\n", u.v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {
    struct var {
        unsigned short v1:4;
        unsigned short v2:7;
    } v;
```



```
    unsigned short p;  
} u;  
  
u.p = -5;  
printf("%d\n", u.v.v2);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {  
    struct var {  
        unsigned short v1:4;  
        unsigned short v2:7;  
        unsigned short v3:4;  
        unsigned short v4:1;  
    } v;  
    unsigned short p;  
} u;  
  
u.p = -16;  
printf("%d, %d\n", u.v.v4, u.v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {  
    struct var {  
        unsigned short v1:4;  
        unsigned short v2:7;  
        unsigned short v3:4;  
        unsigned short v4:1;  
    } v;  
    unsigned short p;  
} u;  
  
u.p = -16;  
printf("%d, %d\n", u.v.v4, u.v.v2);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {  
    struct var {  
        unsigned short v1:4;  
        unsigned short v2:7;  
        unsigned short v3:4;  
        unsigned short v4:1;  
    } v;
```

```
    unsigned short p;  
} u;  
  
u.p = -16;  
printf("%d, %d\n", u.v.v2, u.v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {  
    struct var {  
        unsigned short v1:4;  
        unsigned short v2:7;  
        unsigned short v3:4;  
        unsigned short v4:1;  
    } v;  
    unsigned short p;  
} u;  
  
u.p = -16;  
printf("%d, %d\n", u.v.v3, u.v.v1);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {  
    struct var {  
        unsigned short v1:4;  
        unsigned short v2:7;  
        unsigned short v3:4;  
        unsigned short v4:1;  
    } v;  
    unsigned short p;  
} u;  
  
u.v.v1 = 4;  
u.v.v2 = 3;  
u.v.v3 = 2;  
u.v.v4 = 1;  
  
printf("%d\n", u.p);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
union {  
    struct var {  
        unsigned short v1:4;
```

```
    unsigned short v2:7;
    unsigned short v3:4;
    unsigned short v4:1;
} v;
signed short p;
} u;

u.v.v1 = 4;
u.v.v2 = 3;
u.v.v3 = 2;
u.v.v4 = 1;

printf("%d\n", u.p);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = 125;
signed char b = 5;
signed char c;
```

```
c = a + b;
printf("c=%d\n", c);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = 125;
signed char b = 5;
signed char c;
```

```
c = (unsigned char)(a + b);
printf("c=%d\n", c);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = 125;
signed char b = 5;
unsigned char c;
```

```
c = (signed char)(a + b);
printf("c=%d\n", c);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = 137;
signed char b = 5;
```

```
signed char c;  
  
c = a + b;  
printf("c=%d\n", c);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = 137;  
signed char b = 5;  
signed char c;  
  
c = (unsigned char)(a + b);  
printf("c=%d\n", c);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = 137;  
signed char b = 5;  
unsigned char c;  
  
c = (signed char)(a + b);  
printf("c=%d\n", c);
```

Який розмір пам'яті потрібний для збереження змінної v?

```
struct var {  
    unsigned short v1:4;  
    unsigned short v2:5;  
    unsigned short v3:6;  
} v;
```

Який розмір пам'яті потрібний для збереження змінної v?

```
struct var {  
    unsigned char v1:4;  
    unsigned char v2:5;  
    unsigned char v3:3;  
} v;
```

Який розмір пам'яті потрібний для збереження змінної v?

```
struct var {  
    unsigned short v1:13;  
    unsigned short v2:5;  
    unsigned short v3:6;
```

```
} v;
```

Який розмір пам'яті потрібний для збереження змінної v?

```
struct var {  
    unsigned char v1:5;  
    unsigned char v2:4;  
    unsigned char v3:3;  
    unsigned char v4:3;  
} v;
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = -120;  
signed char b = -11;  
signed char c;  
  
c = a + b;  
printf("c=%d\n", c);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = -120;  
signed char b = -11;  
signed char c;  
  
c = (unsigned char)(a + b);  
printf("c=%d\n", c);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char a = -120;  
signed char b = -11;  
unsigned char c;  
  
c = (signed char)(a + b);  
printf("c=%d\n", c);
```

Що буде виведено на екран при виконанні фрагмента коду:

```
signed char b = -11;  
unsigned char c;  
  
c = (unsigned char)b;  
printf("c=%d\n", c);
```

|  |
|--|
|  |
| Алгоритму характеризується наступними властивостями:   |
| Властивість алгоритму, яка полягає в тому, що алгоритм повинен мати можливість завершення, називається   |
| Властивість алгоритму, яка полягає в тому, що алгоритм повинен приводити до певного результату (можливо, негативного), називається                 |
| Властивість алгоритму, яка полягає в тому, що алгоритм повинен складатися з конкретних дій, які виконуються за кінцевий проміжок часу, називається |
| Властивість алгоритму, яка полягає в тому, що один і той же алгоритм можна використовувати з різними вхідними даними, називається                  |
| Властивість алгоритму яка полягає в тому, що будь-яка дія має бути чітко і недвозначно визначена в кожному випадку, називається                    |
| Алгоритму характеризується наступними властивостями:   |
| В чому полягає така властивість як скінченність?   |
| В чому полягає така властивість як результативність?   |
| В чому полягає така властивість як детермінованість?   |
| В чому полягає така властивість як універсальність?  |
| В чому полягає така властивість як дискретність?   |
| Що можна вважати алгоритмом?   |
| Алгоритм у вигляді графічних символів називається  |
| Який аналіз використовується для оцінки обчислювальної складності алгоритму?   |
| Яка оцінка часу виконання алгоритму з  |

|   |
|---|
| вказаних найбільш ефективна:  |
| Що можна вважати алгоритмом?  |
| <p>Яка асимптотична складність представленого коду?</p> <pre> int i, j, n = 3, r = 0; for(i = 0; i &lt; n; i++) {     for(j = 0; j &lt; n; j++) {         r += i * j;     } } </pre>      |
| <p>Яка асимптотична складність представленого коду?</p> <pre> int i, n = 3, r = 1; for(i = 0; i &lt; n; i++) {     r *= 2 * i; } </pre>   |
| Яка асимптотична складність перебору елементів одномірному масиві?  |
| Яка асимптотична складність перебору елементів двумірному масиві?   |
| Яка асимптотична складність перебору елементів тримірному масиву?   |
| Яка асимптотична складність отримання значення елемента одномірного масиву?   |
| Яка асимптотична складність отримання значення елемента двумірному масиву?  |
| Яка асимптотична складність отримання значення елемента тримірному масиву?  |
| <p>Яка асимптотична складність представленого коду?</p> <pre> int i, j, n = 128, r = 0; for(i = 0; i &lt; n; i++) {     for(j = 1; j &lt; n; j *= 2) {         r += i * j;     } } </pre> |
| Оцінка часу виконання алгоритму бінарного пошуку елемента у відсортованому масиві рівна:  |

|  |
|--|
| Оцінка часу виконання алгоритму лінійного пошуку елемента у масиві рівна:  |
| Який елемент відшукається найшвидше за допомогою алгоритму бінарного пошуку ( $n$ — непарне, $n \geq 3$ )?   |
| Скільки разів поділ масиву потрібно зробити, щоб знайти за допомогою алгоритму бінарного пошуку перший елемент, якщо відомо, що масив містить 11 елементів?              |
| Максимум скільки разів поділ масиву потрібно зробити, щоб знайти за допомогою алгоритму бінарного пошуку будь-який елемент, якщо відомо, що масив містить 15 елементів?  |
| Скільки разів поділ масиву потрібно зробити, щоб знайти за допомогою алгоритму бінарного пошуку останній елемент, якщо відомо, що масив містить 14 елементів?            |
| Яку максимальну кількість ітерацій потрібно виконати, щоб знайти за допомогою алгоритму бінарного пошуку будь-який елемент, якщо відомо, що масив містить $N$ елементів? |
| Скільки разів поділ масиву потрібно зробити, щоб знайти за допомогою алгоритму бінарного пошуку середній елемент, якщо відомо, що масив містить 13 елементів?            |
| Нехай задано масив $M$ (0, 1, 4, 5, 7, 9, 11, 15, 20). Використовуючи алгоритм бінарного пошуку, який з елементів відшукається швидше?                                   |
| Нехай задано масив $M$ (0, 1, 4, 5, 7, 9, 11, 15, 20). Використовуючи алгоритм бінарного пошуку, який з елементів відшукається швидше?                                   |
| Скільки операцій порівняння здійснюється   |



в алгоритмі бінарного пошуку для відшукування елемента, значення якого рівне 4 в масиві  $M(0, 1, 4, 5, 7, 9, 11, 15, 20)$ ?

Яку кількість ітерацій потрібно виконати в алгоритмі лінійного пошуку для відшукування елемента, значення якого рівне 9 в масиві  $M(0, 1, 4, 5, 7, 9, 11, 15, 20)$ ?

Яку кількість ітерацій потрібно виконати в алгоритмі бінарного пошуку для відшукування елемента, значення якого рівне 9 в масиві  $M(0, 1, 4, 5, 7, 9, 11, 15, 20)$ ?

Дан масив розмірністю  $K$ , який містить числа в діапазоні від 0 до  $N$ , де  $N$  значно менше за  $K$ . Яку мінімальну кількість ітерацій необхідно виконати, щоб визначити імовірність появи кожного значення?

Динамічна структура даних в інформатиці, що складається з вузлів, кожен з яких містить як власне дані, так одне або два посилання на наступний і/або попередній вузол зветься

Динамічна структура даних, що представляє собою упорядкований набір елементів, в яку додавання і видалення елементів проводиться з одного боку зветься

Динамічна структура даних, що представляє собою упорядкований набір елементів, в яку додавання елементів проводиться з одного боку, а видалення з іншого боку зветься

Динамічна структура даних в інформатиці, що складається з вузлів, кожен з яких містить як власне дані, так і посилання на один/два аналогічних сусідніх вузла зветься

Що таке зв'язний список?

Що таке стек?

Що таке черга?

Що таке бінарне дерево?

Які зв'язні списки не існують?

Які позиції стека доступні для занесення нових елементів?

|  |
|--|
|  |
| Які позиції стека доступні для вилучення елементів?  |
| Які позиції черги доступні для занесення нових елементів?  |
| Які позиції черги доступні для вилучення елементів?  |
| Яка асимптотична складність отримання значення елемента зі стеку?  |
| Яка асимптотична складність отримання значення елемента з черги?   |
| Яка асимптотична складність отримання значення будь-якого елемента зі зв'язного списку?  |
| Яка асимптотична складність пошуку елемента у зв'язном списку?   |
| Які операції над елементами характерні для черг та стеків?   |
| Перегляд списку дає наступний результат: 2, 4, 6, 8, 10, 12. Відомо, що дані вносилися в початок списку. Як вони були впорядковані перед введенням в список?<br><br>за зменшенням<br>дані були впорядковані випадково<br><br>за зростанням |
| Граф називається повним, якщо:   |
| Неорієнтованим графом називають:   |
| Орієнтованим графом називають:   |
| Шляхом в орієнтованому графі називають:  |
| Простим циклом в орієнтованому графі називають:  |
| Яка асимптотична складність для бульбашкового сортування?  |

|  |
|--|
|  |
| Яка асимптотична складність для сортування вибором?  |
| Яка асимптотична складність для сортування вставками?  |
| Яка асимптотична складність для сортування Шеллом?   |
| Є бінарне дерево пошуку, що містить цілі числа від 1 до 7. Яким буде результат спадного перегляду? |
| Яка структура даних використовується для буферизації введення/виведення?                           |