

1. Які з оголошених нижче змінних є вказівниками?

`int ad0, c1, *ad, *sh;`
`float* m1, m2, *rs;`

2. Які значення можна присвоювати вказівнику? Наведіть приклади.

3. Яким буде значення змінної x після виконання наступного фрагмента програми?

```
int x, *vk;
vk = &x;
x = 5;
*vk = x * 12;
x += *vk;
printf("\n %i", x);
```

4. Які операції можна виконувати над вказівниками? Яким вимогам повинні відповідати вказівники, що вступають у ці операції?

5. Що відбувається у разі виконання операції інкремента/декремента вказівника?

6. Чому дорівнює різниця значень вказівника?

7. На скільки байт зміниться (збільшиться, зменшиться) значення при виконанні наступних операцій з вказівниками?

```
int* vk; vk = vk + 1; _____
char* pch; pch = pch + 73; _____
float* m1; m1-- _____
```

8. У програмі виконано наступні оголошення:

```
char* pch;
void* pt;
int* d, cbr;
```

Які з наступних операторів присвоєння правильні а які помилкові?

<code>pch = &cbr; _____</code>	<code>pt = &cbr; _____</code>
<code>d = &cbr; _____</code>	<code>pt = pch + cbr; _____</code>
<code>pch = (int*)pt; _____</code>	<code>cbr *= *d; _____</code>
<code>cbr = 3*d; _____</code>	<code>cbr = *pt + 8; _____</code>
<code>cbr = *(int *)pt - 8; _____</code>	<code>*d = cbr ** pch; _____</code>

9. Що виведе на екран наступна програма? _____

```
#include <stdio.h>
int main(void)
{
    int ref [ ] = {8, 4, 0, 2} ;
    int *ptr;
    int index;
    for (index = 0, ptr = ref; index < 4; index++, ptr++)
        printf("%d %d\n", ref[index], *ptr) ;
    return 0;
}
```

Чи є ref адресою? Якщо ref адреса то що знаходиться за цією адресою? Скільки елементів містить масив ref?

1. Які з оголошених нижче змінних є вказівниками?

`int ad0, c1, * ad, sh;`
`float* m1, m2, * rs;`

2. Які значення можна присвоювати вказівнику? Наведіть приклади.

3. Яким буде значення змінної x після виконання наступного фрагмента програми?

```
int x, * vk;
vk = &x;
x = 5;
* vk = x * 5;
x *= * vk;
printf("\n %i", x);
```

4. Які операції можна виконувати над вказівниками? Яким вимогам повинні відповідати вказівники, що вступають у ці операції?

5. Що відбувається у разі виконання операції інкремента/декремента вказівника?

6. Чому дорівнює різниця значень вказівника?

7. На скільки байт зміниться (збільшиться, зменшиться) значення при виконанні наступних операцій з вказівниками?

```
int* vk; vk = vk - 2; _____
char* pch; pch = pch + 14; _____
float* m1; m1++; _____
```

8. У програмі виконано наступні оголошення:

```
char* pch;
void* pt;
int* d , cbr;
```

Які з наступних операторів присвоєння правильні а які помилкові?

<code>pch = &cbr; _____</code>	<code>pt = &cbr; _____</code>
<code>d = &cbr; _____</code>	<code>pt = pch + cbr; _____</code>
<code>pch = (int*)pt; _____</code>	<code>cbr *= *d; _____</code>
<code>cbr = 3*d; _____</code>	<code>cbr = *pt + 8; _____</code>
<code>cbr = *(int *)pt - 8; _____</code>	<code>* d = cbr ** pch; _____</code>

9. Що виведе на екран наступна програма? _____

```
#include <stdio.h>
int main(void)
{
    int ref [ ] = {8, 4, 0, 2} ;
    int *ptr;
    int index;
    for (index = 0, ptr = ref; index < 4; index++, ptr++)
        printf("%d %d\n", ref[index], *ptr) ;
    return 0;
}
```

Чи є ref адресою? Що означає ref + 1? На що вказує ++ref?

Український католицький університет
Кафедра комп'ютерних наук та інформаційних технологій
Білет № __ по курсу Основи програмування
Прізвище студента _____ Оцінка _____

1. Які з оголошених нижче змінних є вказівниками?

`Int*` ad0, * c1, ad, sh;
`float*` m1, * m2, rs;

2. Які значення можна присвоювати вказівнику? Наведіть приклади.

3. Яким буде значення змінної x після виконання наступного фрагмента програми?

```
int x, * vk;  
vk = &x;  
x = 5;  
* vk = x + 5;  
x -= * vk;  
printf("\n %i", x);
```

4. Які операції можна виконувати над вказівниками? Яким вимогам повинні відповідати вказівники, що вступають у ці операції?

5. Що відбувається у разі виконання операції інкремента/декремента вказівника?

6. Чому дорівнює різниця значень вказівника?

7. На скільки байт зміниться (збільшиться, зменшиться) значення при виконанні наступних операцій з вказівниками?

<code>int*</code> vk; vk = vk - 7;	_____ - 7*4(бо int) = -28
<code>char*</code> pch; pch = pch + 3;	_____
<code>float*</code> m1; m1++	_____

8. У програмі виконано наступні оголошення:

```
char* pch;  
void* pt;  
int* d , cbr;
```

Які з наступних операторів присвоєння правильні а які помилкові?

pch = &cbr; _____	pt = &cbr; _____
d = &cbr; _____	pt = pch + cbr; _____
pch = (<code>int*</code>)pt; _____	cbr *= *d; _____
cbr = 3*d; _____	cbr = *pt + 8; _____
cbr = *(<code>int*</code>)pt - 8; _____	* d = cbr * * pch; _____

9. У програмі виконано наступні оголошення:

```
float rootbeer[10], *pf, value = 2.2;
```

```
int i = 3;
```

Які з наступних операторів присвоєння правильні а які помилкові?

rootbeer [2] = value; _____	printf("%f", rootbeer); _____
scanf("%f", &rootbeer); _____	pf = value; _____
rootbeer = value; _____	pf = rootbeer; _____

Український католицький університет
Кафедра комп'ютерних наук та інформаційних технологій
Білет № __ по курсу Основи програмування
Прізвище студента _____ Оцінка _____

1. Які з оголошених нижче змінних є вказівниками?

`int ad0, c1, *ad, sh;`
`float* m1, m2, *rs;`

2. Які значення можна присвоювати вказівнику? Наведіть приклади.

3. Яким буде значення змінної x після виконання наступного фрагмента програми?

```
int x, *vk;
vk = &x;
x = 5;
*vk = x * 5;
x += *vk;
printf("\n %i", x);
```

4. Які операції можна виконувати над вказівниками? Яким вимогам повинні відповідати вказівники, що вступають у ці операції?

5. Що відбувається у разі виконання операції інкремента/декремента вказівника?

6. Чому дорівнює різниця значень вказівника?

7. На скільки байт зміниться (збільшиться, зменшиться) значення при виконанні наступних операцій з вказівниками?

```
int* vk; vk = vk + 7; _____
char* pch; pch = pch - 3; _____
float* m1; m1-- _____
```

8. У програмі виконано наступні оголошення:

```
char* pch;
void* pt;
int* d, cbr;
```

Які з наступних операторів присвоєння правильні а які помилкові?

<code>pch = &cbr; _____</code>	<code>pt = &cbr; _____</code>
<code>d = &cbr; _____</code>	<code>pt = pch + cbr; _____</code>
<code>pch = (int*)pt; _____</code>	<code>cbr *= *d; _____</code>
<code>cbr = 3*d; _____</code>	<code>cbr = *pt + 8; _____</code>
<code>cbr = *(int *)pt - 8; _____</code>	<code>*d = cbr ** pch; _____</code>

9. У програмі виконано наступні оголошення:

```
float rootbeer[10], *pf, value = 2.2;
```

```
int i = 3;
```

Які з наступних операторів присвоєння правильні а які помилкові?

<code>rootbeer [2] = value; _____</code>	<code>printf("%f", rootbeer); _____</code>
<code>scanf("%f", &rootbeer); _____</code>	<code>pf = value; _____</code>
<code>rootbeer = value; _____</code>	<code>pf = rootbeer; _____</code>