

Міністерство освіти і науки України  
Національний технічний університет України «Київський політехнічний  
інститут імені Ігоря Сікорського»  
Факультет інформатики та обчислювальної техніки  
Кафедра інформатики та програмної інженерії

Звіт

З лабораторної роботи №5 з дисципліни

“Алгоритми та структури даних-1.  
Основи алгоритмізації ”

«Дослідження складних циклічних алгоритмів»

Варіант 27

Виконав студент ІІІ-11 Савенко Олексій Андрійович  
(шифр, прізвище, ім'я, по батькові)

Перевірів Мартінова О.П.  
( прізвище, ім'я, по батькові)

Київ 2021

## Лабораторна робота 5

### Дослідження складних циклічних алгоритмів

**Мета** – дослідити особливості роботи складних циклів та набутти практичних навичок їх використання під час складання програмних специфікацій.

### Варіант 27

#### Індивідуальне завдання

27. Знайти всі числа, що представляють собою повторювані фрагменти послідовності цифр (наприклад, 3434, 23452345) з інтервалу [1000,1000000]

#### Постановка завдання

Результатом завдання є знаходження усіх відповідних чисел виду  **$n_1n_2n_3...n_1n_2n_3...$** , які є вірними за умовою завдання. Головним інструментом мого алгоритму буде зовнішній арифметичний цикл з початковим значенням лічильника  **$i=1000$**  і кроком  **$i++$** , виконуватися він буде поки значення  **$i$**  не буде дорівнювати **10000000**.

Нам потрібно визначити кількість знаків у числі, знаходитись це буде за допомогою внутрішнього ітераційного циклу він буде виконуватись поки ( **$sfc > 0$** ) для цього збережемо значення  **$i$**  у  **$sfc$** , у циклі кожну ітерацію виконується послідовне цілочисельне ділення  **$sfc / 10$** , перед цим до значення покажчика  **$cofdig$**  додається +1, у кінці ми отримаємо кількість цифр у числі.

В залежності від кількості цифр у числі, до нього буде застосовано певну умову перевірки чи відповідає даний елемент умові завдання, а отже належить до чисел які ми шукаємо, якщо число знайдено ми виводимо його значення  **$i$**  додаємо до лічильника  **$amnseq + 1$** . Після завершення арифметичного циклу, відбувається виведення  **$amnseq$**  з кількістю усіх чисел у даному проміжку, що відповідають умові завдання.

### Математична модель

Змінна	Тип	Ім'я	Призначення
Лічильник арифметичного циклу <b>i</b>	Натуральний	<b>i</b>	Перебір чисел заданого проміжку
Лічильник <b>cofdig</b>	Натуральний	<b>cofdig</b>	Визначення кількості цифр у числі
Отримувач значення <b>sfc</b>	Цілий	<b>sfc</b>	Збереження значення змінної <b>i</b> для ітераційного циклу з визначення знаків
Лічильник <b>amnseq</b>	Цілий	<b>amnseq</b>	Визначення кількості чисел, які відповідають умові завдання
Повернення залишку від ділення	Арифметична дія	<b>%</b>	Ділення з взяттям остачі від ділення
Цілочисельне ділення	Арифметична дія	<b>/</b>	Ділення з взяттям цілої частини від ділення
Рівність	Арифметична дія	<b>==</b>	Позначення рівності значень та їх порівняння

**Крок 1.** Визначимо основні дії

**Крок 2.** Ініціалізація змінної **amnseq**

**Крок 3.** Деталізація арифметичного циклу для перебору чисел послідовності

**Крок 4.** Ініціалізація змінної отримувача **sfc** та лічильника **cofdig**

**Крок 5.** Деталізація ітераційного циклу для знаходження кількості цифр у числі **i**

**Крок 6.** Деталізація умови для 4-знакового числа

**Крок 7.** Деталізація умови для 5-знакового числа

**Крок 8.** Деталізація умови для 6-знакового числа

### **Псевдокод алгоритму**

**Крок 1.**

**Початок**

Ініціалізація змінної amnseq

Деталізація арифметичного циклу для перебору чисел послідовності

Ініціалізація змінної отримувача sfc та лічильника cofdig

Деталізація ітераційного циклу для знаходження кількості знаків у числі i

Деталізація умови для 4-знакового числа

Деталізація умови для 5-знакового числа

Деталізація умови для 6-знакового числа

**Виведення** кількості чисел amnseq

**Кінець**

**Крок 2.**

**Початок**

amnseq = 0

Деталізація арифметичного циклу для перебору чисел послідовності

Ініціалізація змінної отримувача sfc та лічильника cofdig

Деталізація ітераційного циклу для знаходження кількості знаків у числі i

Деталізація умови для 4-знакового числа

Деталізація умови для 5-знакового числа

Деталізація умови для 6-знакового числа

**Виведення** кількості чисел `amnseq`

**Кінець**

**Крок 3.**

**Початок**

`amnseq = 0`

**повторити**

**для  $i$  від 1000 до 1000000 із кроком 1**

Ініціалізація змінної отримувача `sfc` та лічильника `cofdig`

Деталізація ітераційного циклу для знаходження кількості знаків у числі  $i$

Деталізація умови для 4-знакового числа

Деталізація умови для 5-знакового числа

Деталізація умови для 6-знакового числа

**Виведення** кількості чисел `amnseq`

**Кінець**

**Крок 4.**

**Початок**

amnseq = 0

**повторити**

**для i від 1000 до 1000000 із кроком 1**

sfc = i

cofdig = 0

Деталізація ітераційного циклу для знаходження кількості знаків у числі i

Деталізація умови для 4-знакового числа

Деталізація умови для 5-знакового числа

Деталізація умови для 6-знакового числа

**Виведення** кількості чисел amnseq

**Кінець**

**Крок 5.**

**Початок**

amnseq = 0

**повторити**

**для i від 1000 до 1000000 із кроком 1**

sfc = i

cofdig = 0

**поки** sfc > 0

**повторити**

cofdig=cofdig+1

sfc = sfc / 10

**все повторити**

Деталізація умови для 4-значового числа

Деталізація умови для 5-значового числа

Деталізація умови для 6-значового числа

**Виведення** кількості чисел amnseq

**Кінець**

**Крок 6.**

**Початок**

amnseq = 0

**повторити**

**для i від 1000 до 1000000 із кроком 1**

sfc = i

cofdig = 0

**поки** sfc > 0

**повторити**

cofdig=cofdig+1

sfc = sfc / 10

**все повторити**

**Якщо** cofdig == 4 та i / 100 == i % 100

**то**

amnseq++

**Виведення** числа i

**інакше**

Деталізація умови для 5-значового числа

Деталізація умови для 6-значового числа

**Виведення** кількості чисел amnseq

**Кінець**

**Крок 7.**

**Початок**

amnseq = 0

**повторити**

для i від 1000 до 1000000 із кроком 1

sfc = i

cofdig = 0

**поки** sfc > 0

**повторити**

cofdig=cofdig+1

sfc = sfc / 10

**все повторити**

**Якщо** cofdig == 4 та i / 100 == i % 100

**то**

amnseq++

**Виведення** числа i

**інакше**

**Якщо** cofdig == 5 та i % 11111 == 0

**то**

amnseq++

**Виведення** числа i



**інакше**

Деталізація умови для 6-значового числа

**Виведення** кількості чисел amnseq

**Кінець**

**Крок 8.**

**Початок**

amnseq = 0

**повторити**

**для i від 1000 до 1000000 із кроком 1**

sfc = i

cofdig = 0

**поки** sfc > 0

**повторити**

cofdig=cofdig+1

sfc = sfc / 10

**все повторити**

**Якщо** cofdig == 4 та i / 100 == i % 100

**то**

amnseq++

**Виведення** числа i

**інакше**

**Якщо** cofdig == 5 та i % 11111 == 0

**то**

amnseq++

**Виведення** числа i

**інакше**

**Якщо**  $\text{cofdig} == 6$  та  $i \% 1000 == i / 1000$  або  $\text{cofdig} == 6$  та  $i / 10000 == (i \% 10000) / 100$  та  $i / 10000 == (i \% 10000) \% 100$  та  $(i \% 10000) / 100 == (i \% 10000) \% 100$

**то**

$\text{amnseq}++$

**Виведення** числа  $i$

**інакше**

**все повторити**

**Виведення** кількості чисел  $\text{amnseq}$

**Кінець**

**Блок-схема алгоритму:**

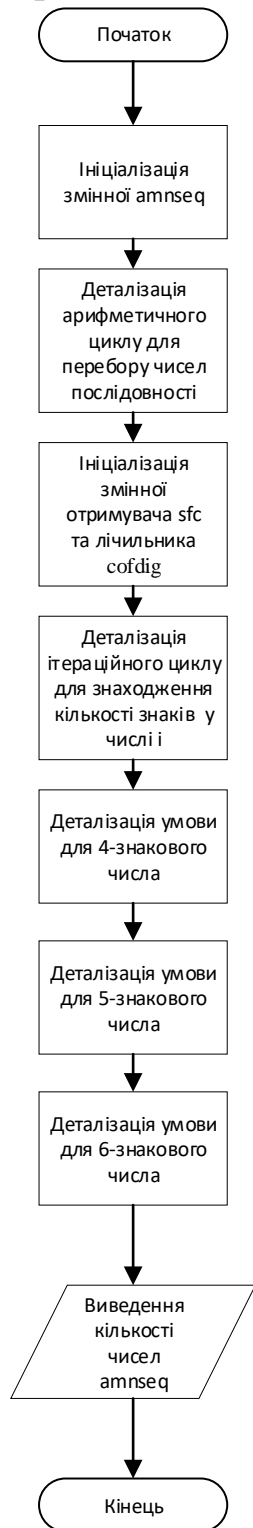
**Легенда блок-схеми**

**Умова 4** -  $\text{cofdig} == 4$  та  $i / 100 == i \% 100$

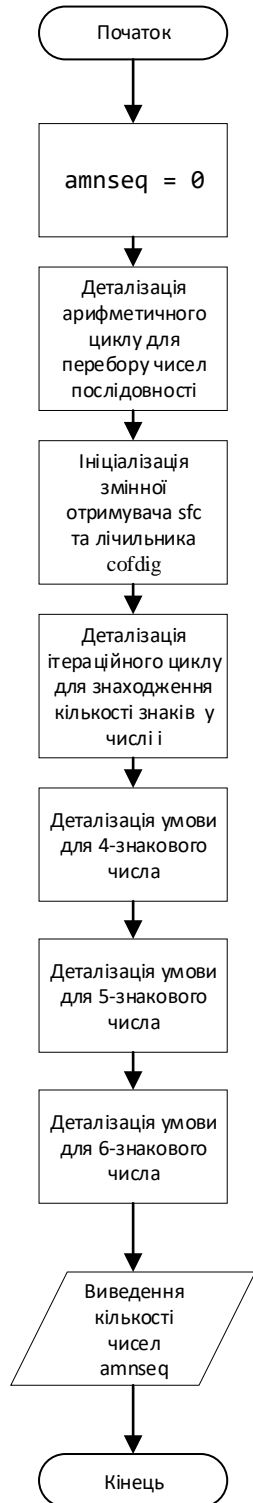
**Умова 5** -  $\text{cofdig} == 5$  та  $i \% 11111 == 0$

**Умова 6** -  $\text{cofdig} == 6$  та  $i \% 1000 == i / 1000$  або  $\text{cofdig} == 6$  та  $i / 10000 == (i \% 10000) / 100$  та  $i / 10000 == (i \% 10000) \% 100$  та  $(i \% 10000) / 100 == (i \% 10000) \% 100$

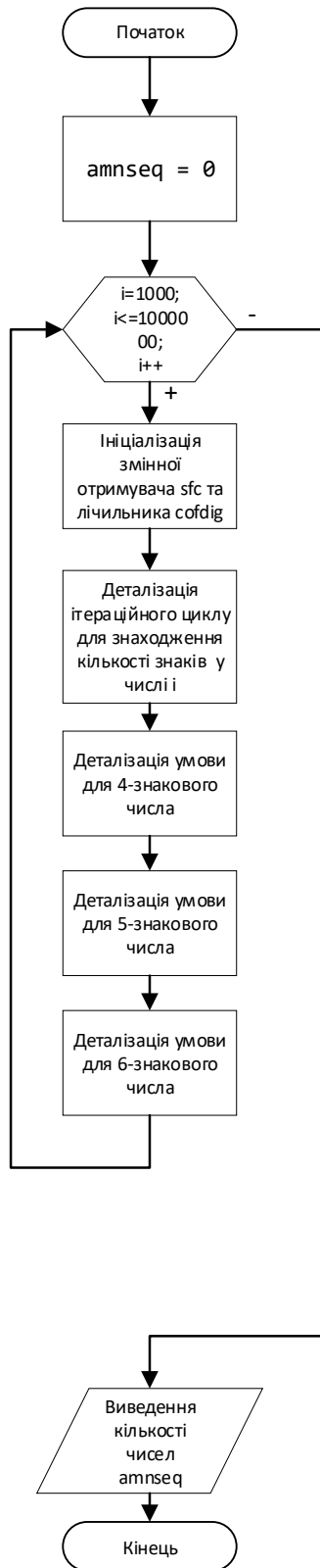
## Крок 1.



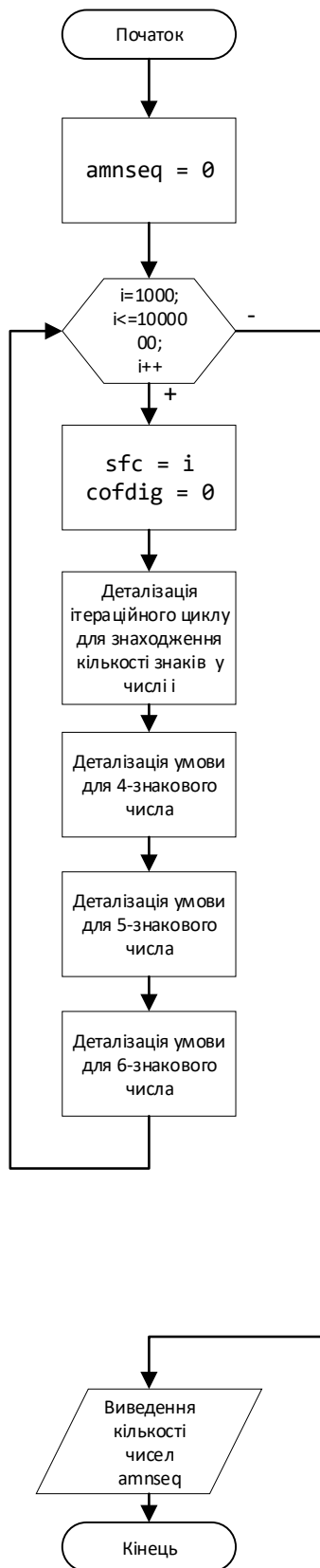
## Крок 2.



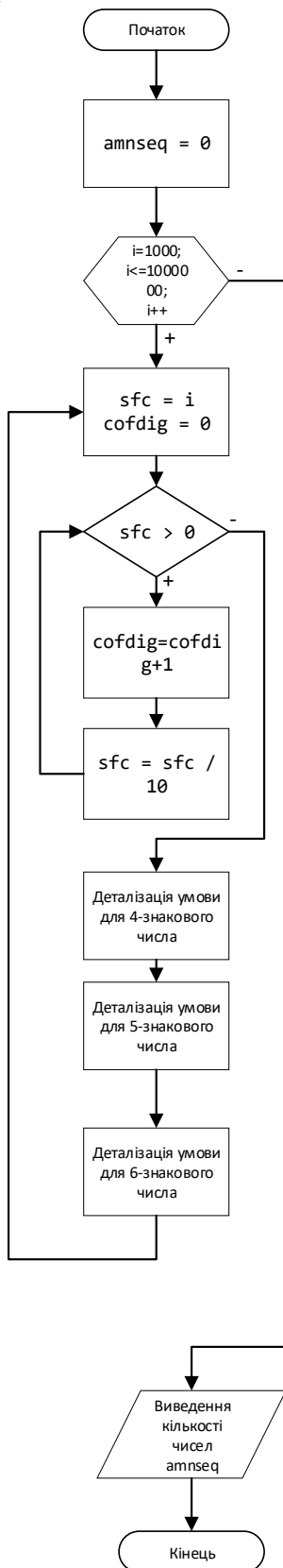
### Крок 3.



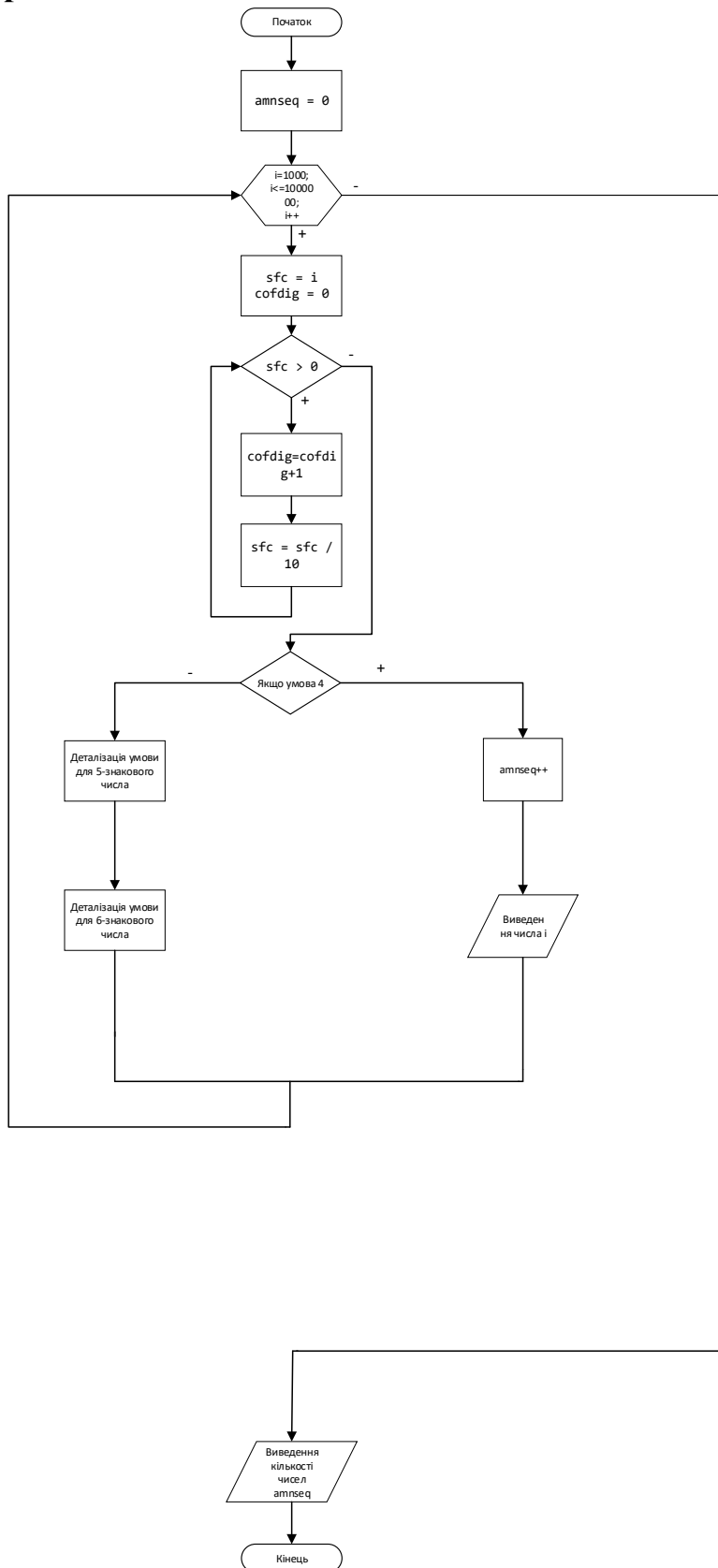
#### Крок 4.



## Крок 5.

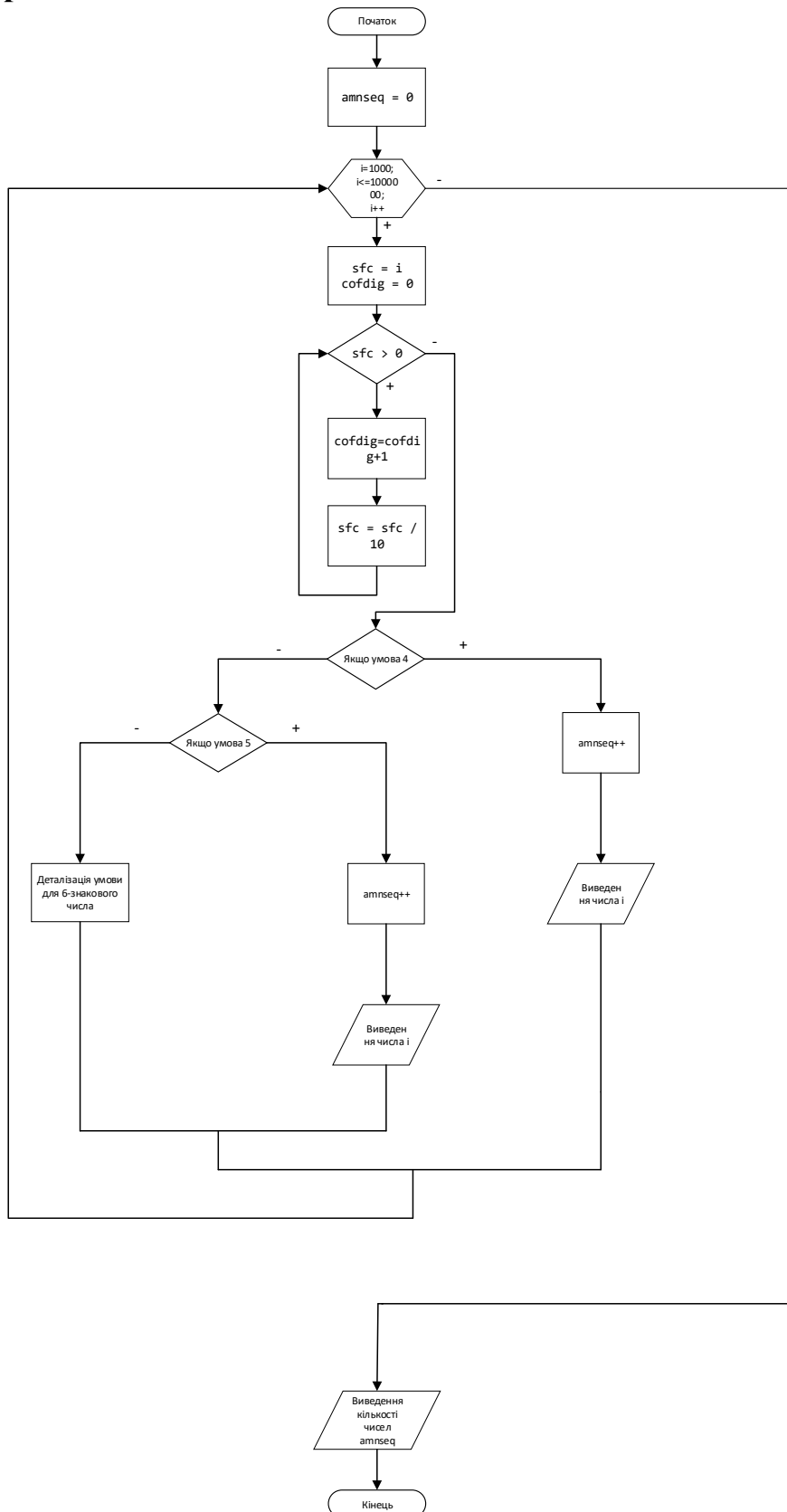


## Крок 6.

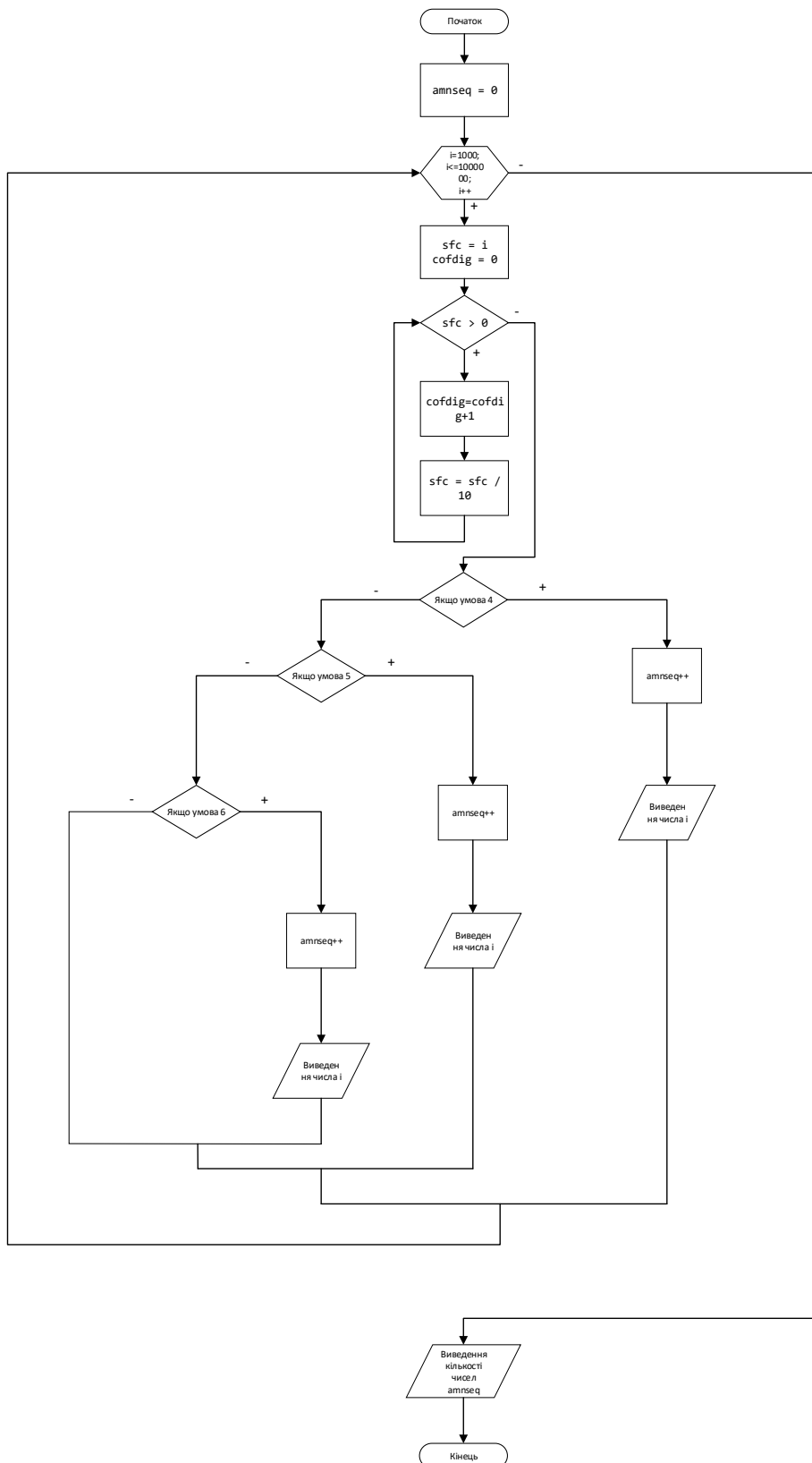




## Крок 7.



## Крок 8.



### Випробування алгоритму

Блок	Дія
	<b>Початок</b>
<b>1</b>	Ініціалізація $amnseq = 0$
<b>2 <math>i=1000; i \leq 1000000; i++</math> (Перевірка умови арифметичного циклу)</b>	$1000 \leq 1000000 == \text{true}$ (Вхід до арифметичного циклу)
<b>3</b>	$sfc = i = 1000$ $cofdig = 0$
<b>4 <math>sfc &gt; 0</math> (Перевірка умови ітераційного циклу)</b>	$1000 > 0 == \text{true}$ (Вхід до ітераційного циклу)
<b>5</b>	$cofdig = cofdig + 1$ $cofdig = 0 + 1 = 1$
<b>6</b>	$sfc = sfc / 10$ $sfc = 1000 / 10 = 100$
<b>7 <math>sfc &gt; 0</math> (Перевірка умови ітераційного циклу)</b>	$100 > 0 == \text{true}$ (Наступна ітерація)
<b>8</b>	$cofdig = cofdig + 1$ $cofdig = 1 + 1 = 2$
<b>9</b>	$sfc = sfc / 10$ $sfc = 100 / 10 = 10$
<b>10 <math>sfc &gt; 0</math> (Перевірка умови ітераційного циклу)</b>	$10 > 0 == \text{true}$ (Наступна ітерація)
<b>11</b>	$cofdig = cofdig + 1$ $cofdig = 2 + 1 = 3$
<b>12</b>	$sfc = sfc / 10$ $sfc = 10 / 10 = 1$
<b>13 <math>sfc &gt; 0</math> (Перевірка умови ітераційного циклу)</b>	$1 > 0 == \text{true}$ (Наступна ітерація)
<b>14</b>	$cofdig = cofdig + 1$ $cofdig = 3 + 1 = 4$
<b>15</b>	$sfc = sfc / 10$ $sfc = 1 / 10 = 0$
<b>16 <math>sfc &gt; 0</math> (Перевірка умови ітераційного циклу)</b>	$0 > 0 == \text{false}$ (Порушення умови, вихід з циклу)
<b>17 Якщо Умова 4</b>	$4 == 4 == \text{true}$ $1000 / 100 = 10$

	$1000 \% 100 = 0$ $10 == 0 == \text{false}$ (Невиконання умови, перехід до дії у випадку її невиконання)
<b>18 Якщо Умова 5</b>	$4 == 5 == \text{false}$ (Невиконання умови, перехід до дії у випадку її невиконання)
<b>19 Якщо Умова 6</b>	$4 == 6 == \text{false}$ (Невиконання умови, перехід до дії у випадку її невиконання (Перехід до наступної ітерації арифметичного циклу))
<b>1</b> <b>i=1111; i&lt;=1000000; i++ (Перевірка умови арифметичного циклу)</b>	$1111 <= 1000000 == \text{true}$ (Наступна ітерація арифметичного циклу)
<b>2</b>	$sfc = i = 1111$ $cofdig = 0$
<b>3 sfc&gt;0 (Перевірка умови ітераційного циклу)</b>	$1111 > 0 == \text{true}$ (Вхід до ітераційного циклу)
<b>4</b>	$cofdig = cofdig + 1$ $cofdig = 0 + 1 = 1$
<b>5</b>	$sfc = sfc / 10$ $sfc = 1111 / 10 = 111$
<b>6 sfc&gt;0 (Перевірка умови ітераційного циклу)</b>	$111 > 0 == \text{true}$ (Наступна ітерація)
<b>7</b>	$cofdig = cofdig + 1$ $cofdig = 1 + 1 = 2$
<b>8</b>	$sfc = sfc / 10$ $sfc = 111 / 10 = 11$
<b>9 sfc&gt;0 (Перевірка умови ітераційного циклу)</b>	$11 > 0 == \text{true}$ (Наступна ітерація)
<b>10</b>	$cofdig = cofdig + 1$ $cofdig = 2 + 1 = 3$
<b>11</b>	$sfc = sfc / 10$ $sfc = 11 / 10 = 1$

<b>12</b> $sfc > 0$ (Перевірка умови ітераційного циклу)	$1 > 0 == \text{true}$ (Наступна ітерація)
<b>13</b>	$\text{cofdig} = \text{cofdig} + 1$ $\text{cofdig} = 3 + 1 = 4$
<b>14</b>	$\text{sfc} = \text{sfc} / 10$ $\text{sfc} = 1 / 10 = 0$
<b>15</b> $sfc > 0$ (Перевірка умови ітераційного циклу)	$0 > 0 == \text{false}$ (Порушення умови, вихід з циклу)
<b>16</b> Якщо Умова 4	$4 == 4 == \text{true}$ $1111 / 100 = 11$ $1111 \% 100 = 11$ $11 == 11 == \text{true}$ (Виконання умови, перехід до дії у випадку її виконання)
<b>17</b>	$\text{amnseq}++$ <b>Виведення</b> числа $i = 1111$ (Перехід до наступної ітерації арифметичного циклу)
<b>1</b> $i = 77777; i \leq 1000000; i++$ (Перевірка умови арифметичного циклу)	$77777 \leq 1000000 == \text{true}$ (Наступна ітерація арифметичного циклу)
<b>2</b>	$\text{sfc} = i = 77777$ $\text{cofdig} = 0$
<b>3</b> $sfc > 0$ (Перевірка умови ітераційного циклу)	$77777 > 0 == \text{true}$ (Вхід до ітераційного циклу)
<b>4</b>	$\text{cofdig} = \text{cofdig} + 1$ $\text{cofdig} = 0 + 1 = 1$
<b>5</b>	$\text{sfc} = \text{sfc} / 10$ $\text{sfc} = 77777 / 10 = 7777$
<b>6</b> $sfc > 0$ (Перевірка умови ітераційного циклу)	$7777 > 0 == \text{true}$ (Наступна ітерація)
<b>7</b>	$\text{cofdig} = \text{cofdig} + 1$ $\text{cofdig} = 1 + 1 = 2$
<b>8</b>	$\text{sfc} = \text{sfc} / 10$ $\text{sfc} = 7777 / 10 = 777$

9 sfc>0 (Перевірка умови ітераційного циклу)	777>0== <b>true</b> (Наступна ітерація)
10	cofdig=cofdig+1 cofdig=2+1=3
11	sfc = sfc / 10 sfc = 777/10=77
12 sfc>0 (Перевірка умови ітераційного циклу)	77>0== <b>true</b> (Наступна ітерація)
13	cofdig=cofdig+1 cofdig=3+1=4
14	sfc = sfc / 10 sfc = 77/10=7
15 sfc>0 (Перевірка умови ітераційного циклу)	7>0== <b>true</b> (Наступна ітерація)
16	cofdig=cofdig+1 cofdig=4+1=5
17	sfc = sfc / 10 sfc = 7/10=0
18 sfc>0 (Перевірка умови ітераційного циклу)	0>0== <b>false</b> (Порушення умови, вихід з циклу)
19 Якщо Умова 4	5==4== <b>false</b> (Невиконання умови, перехід до дії у випадку її невиконання)
20 Якщо Умова 5	5==5== <b>true</b> 77777%11111=0 0==0== <b>true</b> (Виконання умови, перехід до дії у випадку її виконання)
21	amnseq++ <b>Виведення</b> числа i = 77777 (Перехід до наступної ітерації арифметичного циклу)

<b>1</b> <b>i=242424;i&lt;=1000000;i++(Перевірка умови арифметичного циклу)</b>	242424<=1000000== <b>true</b> (Наступна ітерація арифметичного циклу)
<b>2</b>	sfc = i = 242424 cofdig = 0
<b>3 sfc&gt;0 (Перевірка умови ітераційного циклу)</b>	242424>0== <b>true</b> (Вхід до ітераційного циклу)
<b>4</b>	cofdig=cofdig+1 cofdig=0+1=1
<b>5</b>	sfc = sfc / 10 sfc = 242424/10=24242
<b>6 sfc&gt;0 (Перевірка умови ітераційного циклу)</b>	24242>0== <b>true</b> (Наступна ітерація)
<b>7</b>	cofdig=cofdig+1 cofdig=1+1=2
<b>8</b>	sfc = sfc / 10 sfc = 24242/10=2424
<b>9 sfc&gt;0 (Перевірка умови ітераційного циклу)</b>	2424>0== <b>true</b> (Наступна ітерація)
<b>10</b>	cofdig=cofdig+1 cofdig=2+1=3
<b>11</b>	sfc = sfc / 10 sfc = 2424/10=242
<b>12 sfc&gt;0 (Перевірка умови ітераційного циклу)</b>	242>0== <b>true</b> (Наступна ітерація)
<b>13</b>	cofdig=cofdig+1 cofdig=3+1=4
<b>14</b>	sfc = sfc / 10 sfc = 242/10=24
<b>15 sfc&gt;0 (Перевірка умови ітераційного циклу)</b>	24>0== <b>true</b> (Наступна ітерація)
<b>16</b>	cofdig=cofdig+1 cofdig=4+1=5
<b>17</b>	sfc = sfc / 10 sfc = 24/10=2

<b>18</b> $sfc > 0$ (Перевірка умови ітераційного циклу)	$2 > 0 == \text{true}$ (Наступна ітерація)
<b>19</b>	$cofdig = cofdig + 1$ $cofdig = 5 + 1 = 6$
<b>20</b>	$sfc = sfc / 10$ $sfc = 2 / 10 = 0$
<b>21</b> $sfc > 0$ (Перевірка умови ітераційного циклу)	$0 > 0 == \text{false}$ (Порушення умови, вихід з циклу)
<b>22</b> Якщо Умова 4	$6 == 4 == \text{false}$ (Невиконання умови, перехід до дії у випадку її невиконання)
<b>23</b> Якщо Умова 5	$6 == 5 == \text{false}$ (Невиконання умови, перехід до дії у випадку її невиконання)
<b>24</b> Якщо Умова 6	$6 == 6 = \text{true}$ $242424 \% 1000 = 424$ $242424 / 1000 = 242$ $424 == 242 == \text{false}$  $6 == 6 = \text{true}$ $242424 / 10000 = 24$ $(242424 \% 10000) / 100 = 24$ $(242424 \% 10000) \% 100 = 24$ $24 == 24 == 24 == \text{true}$ (Виконання умови, перехід до дії у випадку її виконання)
<b>25</b>	$amnseq++$ <b>Виведення</b> числа $i = 242424$ (Перехід до наступної ітерації арифметичного циклу)
	<b>Виведення</b> $amnseq = 1080$
	<b>Кінець</b>



## Висновок

Отже, я дослідив особливості роботи складних циклів та набути практичних навичок їх використання під час складання програмних специфікацій. У ході вирішення завдання було розроблено алгоритм для його вирішення – визначена постановка завдання, де детально описано логіку алгоритму-основним інструментом якого є **зовнішній арифметичний цикл** головним завданням якого є перебір чисел з проміжку вказаному у завданні, а також **внутрішній ітераційний цикл** для підрахунку кількості знаків у числі, що дозволяє здійснювати відповідні перевірки числа, що до належності їх до елементів, які відповідають умові завдання, в залежності від кількості знаків у ньому, розроблена математична модель – описані відповідні змінні та операції, покроково написаний псевдокод і побудовані блок-схеми. У кінці було проведено випробування алгоритму для перевірки його вірності та можливості застосування для вирішення завдань даного типу – випробування проводилося на 4 числах. **1.1000** було виявлено що число є 4-значовим, що є вірним, а також що воно не відповідає умові завдання, що також є вірним. **2.1111** – число є 4-значовим – вірно, відповідає умові – вірно. **3. 77777** – число є 5-значовим – вірно, відповідає умові - вірно. **4.242424** – число є 6-значовим та відповідає умові завдання, протягом усього арифметичного циклу за рахунок збільшення значення змінної **amnseq** на 1, при знаходженні числа, яке відповідає умові завдання, здійснювався підрахунок кількості чисел, що представляють собою повторювані фрагменти послідовності цифр на проміжку **[1000,1000000]**, усього їх виявилось = 1080.

Таким чином, мій алгоритм є вірним і його можна використовувати для вирішення завдань даного типу.