Міністерство освіти і науки України

**Прикарпатський національний університет**

**імені В.Стефаника**

*Факультет математики та інформатики*

*Кафедра інформаційних технологій*

*Людино-машинна взаємодія*

Лабораторна робота № 5

Тема: Робота з контейнерами в середовищі QtCreator: QVector.

Одновимірні та двовимірні масии.

Варіант: 14

Виконав: ***Пикало М.І***

Група ІПЗ-23

Дата: 15 листопада 2024р.

Викладач: Пікуляк М.В.

Івано-Франківськ - 2024

**Завдання для виконання**

**Варіант 2**

1. Дано цілочисельний масив розміру N. Перетворити його, додавши до

парних чисел перший елемент. Перший і останній елементи масиву не

змінювати.

2. Дано цілочисельний масив розміру N. Вивести спочатку всі його парні

елементи, а потім непарні, зберігаючи порядок проходження елементів

3. Дана матриця розміру m x n. Знайти суми елементів всіх її парних і

непарних стовпців (знаходження суми парних / непарних стовпців

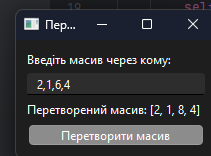
визначається користувачем у віджеті comboBox).

4. Дана матриця розміру m x n. Перетворити матрицю, помінявши місцями

мінімальний і максимальний елемент в кожному рядку.

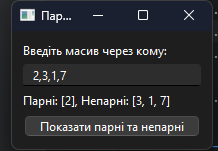
**Завдання 1:**

import sys  
from PySide6.QtWidgets import (  
 QApplication,  
 QWidget,  
 QVBoxLayout,  
 QLabel,  
 QPushButton,  
 QLineEdit,  
)  
  
  
class MainWindow(QWidget):  
 def \_\_init\_\_(self) -> None:  
 super().\_\_init\_\_()  
 self.setWindowTitle("Перетворення масиву")  
  
 self.layout = QVBoxLayout()  
  
 self.input\_label = QLabel("Введіть масив через кому:")  
 self.layout.addWidget(self.input\_label)  
  
 self.input\_array = QLineEdit()  
 self.layout.addWidget(self.input\_array)  
  
 self.result\_label = QLabel("Результат:")  
 self.layout.addWidget(self.result\_label)  
  
 self.transform\_btn = QPushButton("Перетворити масив")  
 self.transform\_btn.clicked.connect(self.transform\_array)  
 self.layout.addWidget(self.transform\_btn)  
  
 self.setLayout(self.layout)  
  
 def transform\_array(self) -> None:  
 array = list(map(int, self.input\_array.text().split(",")))  
 result = [  
 (  
 item  
 if i == 0 or i == len(array) - 1  
 else item + array[0] if item % 2 == 0 else item  
 )  
 for i, item in enumerate(array)  
 ]  
 self.result\_label.setText(f"Перетворений масив: {result}")  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 app = QApplication(sys.argv)  
 window = MainWindow()  
 window.show()  
 sys.exit(app.exec())



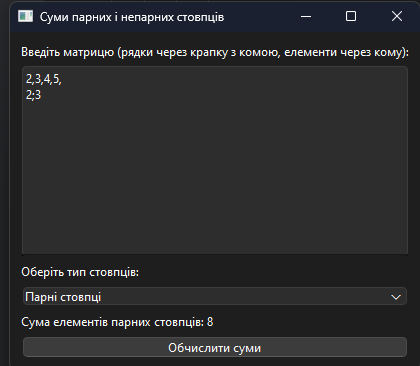
**Завдання 2**

import sys  
from PySide6.QtWidgets import QApplication, QWidget, QVBoxLayout, QLabel, QPushButton, QLineEdit  
  
class MainWindow(QWidget):  
 def \_\_init\_\_(self) -> None:  
 super().\_\_init\_\_()  
 self.setWindowTitle("Парні та непарні елементи масиву")  
  
 self.layout = QVBoxLayout()  
  
 self.input\_label = QLabel("Введіть масив через кому:")  
 self.layout.addWidget(self.input\_label)  
  
 self.input\_array = QLineEdit()  
 self.layout.addWidget(self.input\_array)  
  
 self.result\_label = QLabel("Результат:")  
 self.layout.addWidget(self.result\_label)  
  
 self.show\_btn = QPushButton("Показати парні та непарні")  
 self.show\_btn.clicked.connect(self.show\_even\_odd)  
 self.layout.addWidget(self.show\_btn)  
  
 self.setLayout(self.layout)  
  
 def show\_even\_odd(self) -> None:  
 array = list(map(int, self.input\_array.text().split(',')))  
 even = [x for x in array if x % 2 == 0]  
 odd = [x for x in array if x % 2 != 0]  
 self.result\_label.setText(f"Парні: {even}, Непарні: {odd}")  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 app = QApplication(sys.argv)  
 window = MainWindow()  
 window.show()  
 sys.exit(app.exec())



**Завдання 3**

import sys  
from PySide6.QtWidgets import (  
 QApplication,  
 QWidget,  
 QVBoxLayout,  
 QLabel,  
 QPushButton,  
 QComboBox,  
 QTextEdit,  
)  
  
  
class MainWindow(QWidget):  
 def \_\_init\_\_(self) -> None:  
 super().\_\_init\_\_()  
 self.setWindowTitle("Суми парних і непарних стовпців")  
  
 self.layout = QVBoxLayout()  
  
 self.input\_label = QLabel(  
 "Введіть матрицю (рядки через крапку з комою, елементи через кому):"  
 )  
 self.layout.addWidget(self.input\_label)  
  
 self.input\_matrix = QTextEdit()  
 self.layout.addWidget(self.input\_matrix)  
  
 self.combo\_label = QLabel("Оберіть тип стовпців:")  
 self.layout.addWidget(self.combo\_label)  
  
 self.combo\_box = QComboBox()  
 self.combo\_box.addItems(["Парні стовпці", "Непарні стовпці"])  
 self.layout.addWidget(self.combo\_box)  
  
 self.result\_label = QLabel("Результат:")  
 self.layout.addWidget(self.result\_label)  
  
 self.sum\_btn = QPushButton("Обчислити суми")  
 self.sum\_btn.clicked.connect(self.sum\_columns)  
 self.layout.addWidget(self.sum\_btn)  
  
 self.setLayout(self.layout)  
  
 def sum\_columns(self) -> None:  
 matrix = [  
 list(map(int, row.split(",")))  
 for row in self.input\_matrix.toPlainText().split(";")  
 ]  
  
 selected\_option = self.combo\_box.currentText()  
  
 if selected\_option == "Парні стовпці":  
 sum\_columns = sum(  
 [  
 row[i]  
 for row in matrix  
 for i in range(1, len(row), 2)  
 if i < len(row)  
 ]  
 )  
 self.result\_label.setText(f"Сума елементів парних стовпців: {sum\_columns}")  
 else:  
 sum\_columns = sum(  
 [  
 row[i]  
 for row in matrix  
 for i in range(0, len(row), 2)  
 if i < len(row)  
 ]  
 )  
 self.result\_label.setText(  
 f"Сума елементів непарних стовпців: {sum\_columns}"  
 )  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 app = QApplication(sys.argv)  
 window = MainWindow()  
 window.show()  
 sys.exit(app.exec())



**Завдання 4**

import sys  
from PySide6.QtWidgets import (  
 QApplication,  
 QWidget,  
 QVBoxLayout,  
 QLabel,  
 QPushButton,  
 QTextEdit,  
)  
  
  
class MainWindow(QWidget):  
 def \_\_init\_\_(self) -> None:  
 super().\_\_init\_\_()  
 self.setWindowTitle("Мін. і макс. у рядках матриці")  
  
 self.layout = QVBoxLayout()  
  
 self.input\_label = QLabel(  
 "Введіть матрицю (рядки через крапку з комою, елементи через кому):"  
 )  
 self.layout.addWidget(self.input\_label)  
  
 self.input\_matrix = QTextEdit()  
 self.layout.addWidget(self.input\_matrix)  
  
 self.result\_label = QLabel("Результат:")  
 self.layout.addWidget(self.result\_label)  
  
 self.swap\_btn = QPushButton("Замінити мін. і макс.")  
 self.swap\_btn.clicked.connect(self.swap\_min\_max)  
 self.layout.addWidget(self.swap\_btn)  
  
 self.setLayout(self.layout)  
  
 def swap\_min\_max(self) -> None:  
 matrix = [  
 list(map(int, row.split(",")))  
 for row in self.input\_matrix.toPlainText().split(";")  
 ]  
 for i, row in enumerate(matrix):  
 min\_idx, max\_idx = row.index(min(row)), row.index(max(row))  
 row[min\_idx], row[max\_idx] = row[max\_idx], row[min\_idx]  
 self.result\_label.setText(f"Перетворена матриця: {matrix}")  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 app = QApplication(sys.argv)  
 window = MainWindow()  
 window.show()  
 sys.exit(app.exec())

