

## Pertemuan 7

The screenshot shows a C++ IDE with the file 'quizalgo.cpp' open. The code defines a menu system for a quiz application. It includes headers for `iostream`, `cstdlib`, and `iomanip`, and uses the `std` namespace. The program uses a `pillih` array to store menu options and a `MAX_SIZE` constant. The `menu()` function displays the menu, and `backMenu()` handles returning to the menu. The `tampilkan data array` function shows the contents of an array. The `showdata` function displays data based on the menu selection.

```
1 // Nanang Tr- Nur Wicaksono
2 // 231011700253
3
4 #include <iostream>
5 #include <cstdlib>
6 #include <iomanip>
7 using namespace std;
8
9 int pilli;
10 char pillih[5];
11 const int MAX_SIZE = 30;
12
13 // menu
14 void menu() {
15     system("cls");
16     cout << "=====
17     cout << "Quiz ALGO" << endl;
18     cout << "=====
19     cout << "[1] Input Array 1 Dimensi";
20     cout << "[2] Quick Sort";
21     cout << "[3] Selection Sort";
22     cout << "[4] Insertion Sort";
23     cout << "[5] Keluar";
24     cout << "=====
25     cout << "Pilihan [1 - 5] : ";
26 }
27
28 // back to menu
29 void backMenu() {
30     char status;
31     cout << "Kembali ke menu [y/n] : ";
32     cin >> status;
33     if(status == 'y')
34         menu();
35     else
36         backMenu();
37 }
38
39 // tampilan data array
40 void showData(int arr[], int size) {
41     cout << "[ ";
42     for (int i = 0; i < size; i++) {
43         cout << arr[i];
44         if(i != (size-1))
45             cout << ", ";
46     }
47     cout << " ] ";
48     cout << endl;
49 }
50
51 // tampilan data
52 void showData(int arr[], int arrOld[], int size, int menu) {
53     if(size > 0) {
54         switch (menu) {
55             case 1:
56                 cout << "=====
57                 cout << "Data Array 1 Dimensi : ";
58                 break;
59         }
60     }
61 }
```

The execution output shows the menu displayed in the terminal:

```
=====
Quiz ALGO
=====
[1] Input Array 1 Dimensi
[2] Quick Sort
[3] Selection Sort
[4] Insertion Sort
[5] Keluar
=====
Pilihan [1 - 5] :
```

## Main Menu

The screenshot shows the main menu of the quiz application, titled 'QUIZ ALGO'. The menu options are listed as follows:

```
=====
QUIZ ALGO
=====
[1] Input Array 1 Dimensi
[2] Quick Sort
[3] Selection Sort
[4] Insertion Sort
[5] Keluar
=====
Pilihan [1 - 5] :
```

## Menu 1

```
C:\Users\Nanx\Documents\C++ Unpam\quizalgo.exe

=====
                Input Array 1 Dimensi
=====
Enter the length of the array (not more than 30) : 5
Enter 5 element for the array:
Enter element at Index [0] : 8
Enter element at Index [1] : 3
Enter element at Index [2] : 6
Enter element at Index [3] : 1
Enter element at Index [4] : 7
=====
Data Array 1 Dimensi : [ 8, 3, 6, 1, 7]
Back to menu [y/n] :
```

## Menu 2

```
C:\Users\Nanx\Documents\C++ Unpam\quizalgo.exe

=====
                Quick Sort
=====
Data Array Sebelum Quick Sort : [ 8, 3, 6, 1, 7]
Quick Sort                    : [ 1, 3, 6, 7, 8]
Back to menu [y/n] :
```

## Menu 3

```
C:\Users\Nanx\Documents\C++ Unpam\quizalgo.exe

=====
                        Selection Sort
=====
Data Array Sebelum Selection Sort : [ 8, 3, 6, 1, 7]
Selection Sort                    : [ 1, 3, 6, 7, 8]
Back to menu [y/n] :
```

#### Menu 4

```
C:\Users\Nanx\Documents\C++ Unpam\quizalgo.exe

=====
                        Insertion Sort
=====
Data Array Sebelum Insertion Sort : [ 8, 3, 6, 1, 7]
Insertion Sort                    : [ 1, 3, 6, 7, 8]
Back to menu [y/n] :
```

#### Menun 5

```
CA\Users\Nanx\Documents\C++ Unpam\quizalgo.exe
-- Terimah Kasih --
-----
Process exited after 215.5 seconds with return value 0
Press any key to continue . . .
```

```
// Nanang Tri Nur Wicaksono
// 231011700253
```

```
#include <iostream>
#include <cstdlib>
#include <iomanip>
using namespace std;
```

```
int pil;
char pilihan[5];
const int MAX_SIZE = 30;
```

```
// menu
void menu() {
    system("cls");
    cout << "\n\t=====";
    cout << "\n\t" << setw(25) << "QUIZ ALGO" << setw(25);
    cout << "\n\t=====";
    cout << "\n\t| [1] Input Array 1 Dimensi";
    cout << "\n\t| [2] Quick Sort";
    cout << "\n\t| [3] Selection Sort";
    cout << "\n\t| [4] Insertion Sort";
    cout << "\n\t| [5] Keluar";
    cout << "\n\t=====";
    cout << "\n\tPilihan [1 - 5] : ";
}
```

```

// back to menu
void backMenu(){
    char status;
    cout << "\tBack to menu [y/n] : ";
    cin >> status;
    if(status == 'y')
        menu();
    else
        backMenu();
}

// tampilkan data array
void showArr(int arr[], int size){
    cout << "[ ";
    for (int i = 0; i < size; i++) {
        cout << arr[i];
        if(i != (size-1)){
            cout << ", ";
        }
    }
    cout << "]";
    cout << endl;
}

// tampilkan data
void showData(int arr[], int arrOld[], int size, int menu) {
    if(size > 0){
        switch (menu){
            case 1:
                cout <<
"\t=====\\n";
                cout << "\tData Array 1 Dimensi : ";
                break;
            case 2:
                cout << "\tData Array Sebelum Quick Sort : ";
                showArr(arrOld,size);
                cout << "\tQuick Sort" << setw(22) << " : ";
                break;
            case 3:
                cout << "\tData Array Sebelum Selection Sort : ";
                showArr(arrOld,size);
                cout << "\tSelection Sort" << setw(22) << " : ";
                break;
            case 4:
                cout << "\tData Array Sebelum Insertion Sort : ";
                showArr(arrOld,size);
                cout << "\tInsertion Sort" << setw(22) << " : ";

```

```

                break;
            }
            showArr(arr,size);
        }else
            cout << "\tData Array anda kosong. Silahkan input terlebih dahulu \n";
    }

// menu 1 input data array
void inputArr(int arr[], int arrOld[], int& size){
    system("cls");
    cout << "\n\t=====";
    cout << "\n\t" << setw(35) << "Input Array 1 Dimensi" << setw(35);
    cout << "\n\t=====";
    cout << "\n\tEnter the length of the array (not more than " << MAX_SIZE << ") : ";
    cin >> size;

    if(size <= 0 || size > MAX_SIZE){
        cerr << "\tInvalid array Size. Exiting rthe Program." << endl;
        exit(EXIT_FAILURE);
    }

    cout << "\tEnter " << size << " element for the array: \n";
    for(int i = 0; i < size; i++){
        cout << "\tEnter element at Index [" << i << "] : ";
        cin >> arr[i];
        arrOld[i] = arr[i];
    }
}

// sorting sort
void quickSort(int arr[], int low, int high) {
    system("cls");
    cout << "\n\t=====";
    cout << "\n\t" << setw(32) << "Quick Sort" << setw(32);
    cout << "\n\t===== \n";
    if (low < high) {
        int pivot = arr[high];
        int i = low - 1;
        for (int j = low; j <= high - 1; j++) {
            if (arr[j] < pivot) {
                i++;
                swap(arr[i], arr[j]);
            }
        }
        swap(arr[i + 1], arr[high]);
        int pi = i + 1;
        quickSort(arr, low, pi - 1);
        quickSort(arr, pi + 1, high);
    }
}

```

```

    }
}

// selection sort
void selectionSort(int arr[], int size) {
    system("cls");
    cout << "\n\t=====";
    cout << "\n\t" << setw(35) << "Selection Sort" << setw(35);
    cout << "\n\t=====\\n";
    for (int i = 0; i < size - 1; i++) {
        int minIndex = i;
        for (int j = i + 1; j < size; j++) {
            if (arr[j] < arr[minIndex]) {
                minIndex = j;
            }
        }
        swap(arr[i], arr[minIndex]);
    }
}

// insertion sort
void insertionSort(int arr[], int size) {
    system("cls");
    cout << "\n\t=====";
    cout << "\n\t" << setw(35) << "Insertion Sort" << setw(35);
    cout << "\n\t=====\\n";
    for (int i = 1; i < size; i++) {
        int key = arr[i];
        int j = i - 1;
        while (j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j = j - 1;
        }
        arr[j + 1] = key;
    }
}

// main default
int main() {
    int size;
    int arrOld[MAX_SIZE];
    int arr[MAX_SIZE];
    menu();
    do {
        cin >> pilihan;
        pil = atoi(pilihan);
        switch (pil) {
            case 1:

```

```
        inputArr(arr, arrOld, size);
        showData(arr, arrOld, size, pil);
                backMenu();

        break;

    case 2:
        quickSort(arr, 0, size - 1);
        showData(arr, arrOld, size, pil);
        backMenu();
        break;

    case 3:
        selectionSort(arr,size);
        showData(arr, arrOld, size, pil);
        backMenu();

                break;

    case 4:
        insertionSort(arr,size);
        showData(arr, arrOld, size, pil);
        backMenu();
        break;

    case 5:
        system("cls");
        cout << "\n\t -- Terimah Kasih -- ";
        break;

    default:
        menu();
        cout << "\n\t -- Terimah Kasih -- ";
        break;
    }
} while (pil < 5);

return 0;
}
```



