

NAMA : Yunita Puspitasari

NPM : 242310046

QUIZ ALGORITMA PERTEMUAN 14

1. QUIZ_1

```
1  #include <iostream>
2  #include <algorithm>
3
4  using namespace std;
5
6  const int D = 100;
7  int angka [D] = {11, 22, 33, 44, 55, 66, 77};
8
9  int binarySearch(int x, int l, int r) {
10     if (r >= l) {
11         int mid = l + (r - l) / 2;
12
13         if (angka[mid] == x)
14             return mid;
15
16         if (angka[mid] > x)
17             return binarySearch(x, l, mid - 1);
18
19         return binarySearch(x, mid + 1, r);
20     }
21     return -1;
22 }
23
24 int main() {
25     int n;
26     cout << "Masukkan jumlah angka: ";
27     cin >> n;
28
29     cout << "Masukkan angka : " << endl;
30
31     return -1;
32 }
33
34 int main() {
35     int n;
36     cout << "Masukkan jumlah angka: ";
37     cin >> n;
38
39     cout << "Masukkan angka : " << endl;
40     for (int i = 0; i < n; i++) {
41         cin >> angka[i];
42     }
43
44     sort(angka, angka + n);
45
46     int x;
47     cout << "Masukkan angka yang dicari: ";
48     cin >> x;
49
50     int hasil = binarySearch(x, 0, n - 1);
51
52     if (hasil == -1)
53         cout << "angka tidak ditemukan" << endl;
54     else
55         cout << "angka ditemukan pada indeks " << hasil << endl;
56
57     return 0;
58 }
```

OUTPUT :

```

Masukkan jumlah angka: 7
Masukkan angka :
11
22
33
44
55
66
77
Masukkan angka yang dicari: 44
angka ditemukan pada indeks 3

-----
Process exited after 25.35 seconds with return value 0
Press any key to continue . . .

```

2. QUIZ_2

```

1  #include <iostream>
2  using namespace std;
3
4  struct Node {
5      char data;
6      Node* left;
7      Node* right;
8
9      Node(char val) {
10         data = val;
11         left = right = NULL;
12     }
13 };
14
15 void inorderTraversal(Node* root) {
16     if (root != NULL) {
17         inorderTraversal(root->left);
18         cout << root->data << " ";
19         inorderTraversal(root->right);
20     }
21 }
22
23 int main() {
24     Node* root = new Node('A');
25     root->left = new Node('B');
26     root->right = new Node('C');
27     root->left->left = new Node('D');
28     root->left->right = new Node('E');
29
30     cout << "Inorder Traversal: ";

```

```

8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
Node(char val) {
    data = val;
    left = right = NULL;
}

void inorderTraversal(Node* root) {
    if (root != NULL) {
        inorderTraversal(root->left);
        cout << root->data << " ";
        inorderTraversal(root->right);
    }
}

int main() {
    Node* root = new Node('A');
    root->left = new Node('B');
    root->right = new Node('C');
    root->left->left = new Node('D');
    root->left->right = new Node('E');

    cout << "Inorder Traversal: ";
    inorderTraversal(root);
    cout << endl;

    return 0;
}

```

OUTPUT :

The screenshot shows a Windows command prompt window with the title bar "D:\ALGORITMA SEMS 2\PERTI". The output of the program is displayed as follows:

```

Inorder Traversal: D B E A C

-----
Process exited after 17.18 seconds with return value 0
Press any key to continue . . .

```