

Nama : Andyan Yogawardhana

NIM : 21/482180/PA/21030

Kelas : KOMA

Assignment – 8

1. Membalik Bilangan

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

// deklarasi fungsi prototype reverse
string reverse(string n);

// definisi fungsi main
int main() {
    // deklarasi variabel
    string n;

    // input bilangan yang akan ditukar
    cin >> n;

    // pemanggilan fungsi reverse
    // print hasil penukaran bilangan
    cout << reverse(n);

    return 0;
}

// definisi fungsi reverse
string reverse(string n) {
    // batas rekursi, jika panjang string kurang dari / sama dengan
    1, kembalikan nilai string tersebut ke fungsi
    if(n.length() <= 1) return n;

    // batas rekursi, jika panjang string lebih dari 1, kembalikan
    nilai karakter terakhir dari string input ditambah dengan rekursi
    fungsi reverse dari nilai string pada index pertama hingga index
    terakhir dikurangi satu
    // perulangan akan terus dijalankan hingga panjang string = 1
    // substr(x, y) adalah fungsi untuk mengambil sebagian elemen
    dari sebuah string, dimulai dari index ke-x sampai index ke-y
    else return n.substr(n.length() - 1) + reverse(n.substr(0,
n.length() - 1));
}
```

Nama : Andyan Yogawardhana

NIM : 21/482180/PA/21030

Kelas : KOMA

2. Koefisien Binomial

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

// deklarasi fungsi prototype binomial
int binomial(int n, int k);

// definisi fungsi main
int main() {
    // deklarasi variabel
    int n, k;

    // input variabel indeks
    cin >> n >> k;

    // pemanggilan fungsi binomial
    // print nilai indeks
    cout << binomial(n, k);

    return 0;
}

// definisi fungsi binomial
int binomial(int n, int k) {
    // batas rekursi, jika nilai n = k, n = 1, atau k = 0, kembalikan
    // nilai 1 ke fungsi
    // karena setiap indeks dengan nilai n yang sama dengan k, n = 1
    // atau k = 0 pasti memiliki nilai 1
    if(n == k || n == 1 || k == 0) return 1;

    /* rekursi fungsi binomial
    - nilai dari indeks yang dimasukkan user adalah jumlah dari dua
    angka yang berada di indeks atasnya pada segitiga pascal
    - pola tersebut dilanjutkan terus menerus hingga mencapai
    batasnya, yaitu di puncak segitiga pascal seperti kondisi pada if di
    atas
    - sehingga rumus yang digunakan untuk mencari nilai dari indeks
    adalah rekursi dari fungsi binomial dengan pengurangan nilai n dan k
    sebagai berikut */
    else return binomial(n - 1, k - 1) + binomial(n - 1, k);
}
```

Nama : Andyan Yogawardhana

NIM : 21/482180/PA/21030

Kelas : KOMA

3. Fungsi Ackermann

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

// deklarasi fungsi prototype a (ackermann)
int a(int m, int n);

// definisi fungsi main
int main() {
    // deklarasi variabel
    int m, n;

    // input bilangan parameter
    cin >> m >> n;

    // pemanggilan fungsi a
    // print nilai fungsi
    cout << a(m, n);

    return 0;
}

// deklarasi fungsi prototype a (ackermann)
int a(int m, int n) {
    // batas rekursi, jika nilai m = 0, kembalikan nilai n + 1 ke
    // fungsi (sesuai dengan aturan dari fungsi ackermann)
    if(m == 0) return n + 1;

    // batas rekursi, jika nilai n = 0, kembalikan nilai rekursi
    // fungsi a di mana m dikurangi 1
    else if(n == 0) return a(m - 1, 1);

    // jika m dan n keduanya bukan 0, fungsi rekursi yang digunakan
    // adalah sesuai dengan rumus fungsi ackermann, yaitu sebagai berikut
    else return a(m - 1, a(m, n - 1));
}
```

Nama : Andyan Yogawardhana

NIM : 21/482180/PA/21030

Kelas : KOMA

4. Palindrom

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

// deklarasi fungsi prototype palindrome
string palindrome(string a);

// definisi fungsi main
int main() {
    // deklarasi variabel
    string s;

    // input string
    cin >> s;

    // pemanggilan fungsi palindrome
    // print jenis string (palindrome/bukan)
    cout << palindrome(s);

    return 0;
}

// definisi fungsi palindrome
string palindrome(string s) {
    // jika string hanya terdiri dari 1 karakter, maka string
    tersebut termasuk palindrom
    if(s.length() <= 1) return "Palindrom";

    // jika string terdiri dari 2 karakter atau lebih
    else {
        // jika huruf pertama dari string sama dengan huruf terakhir
        dari string
        // kembalikan nilai string baru namun dimulai dari huruf
        kedua dari awal string dan diakhiri pada huruf kedua dari akhir
        string
        if(s[0] == s[s.length() - 1]) return palindrome(s.substr(1,
s.length() - 2));

        // jika huruf pertama dan huruf terakhir tidak sama, maka
        string otomatis bukan palindrom
        else return "Bukan Palindrom";
    }
}
```

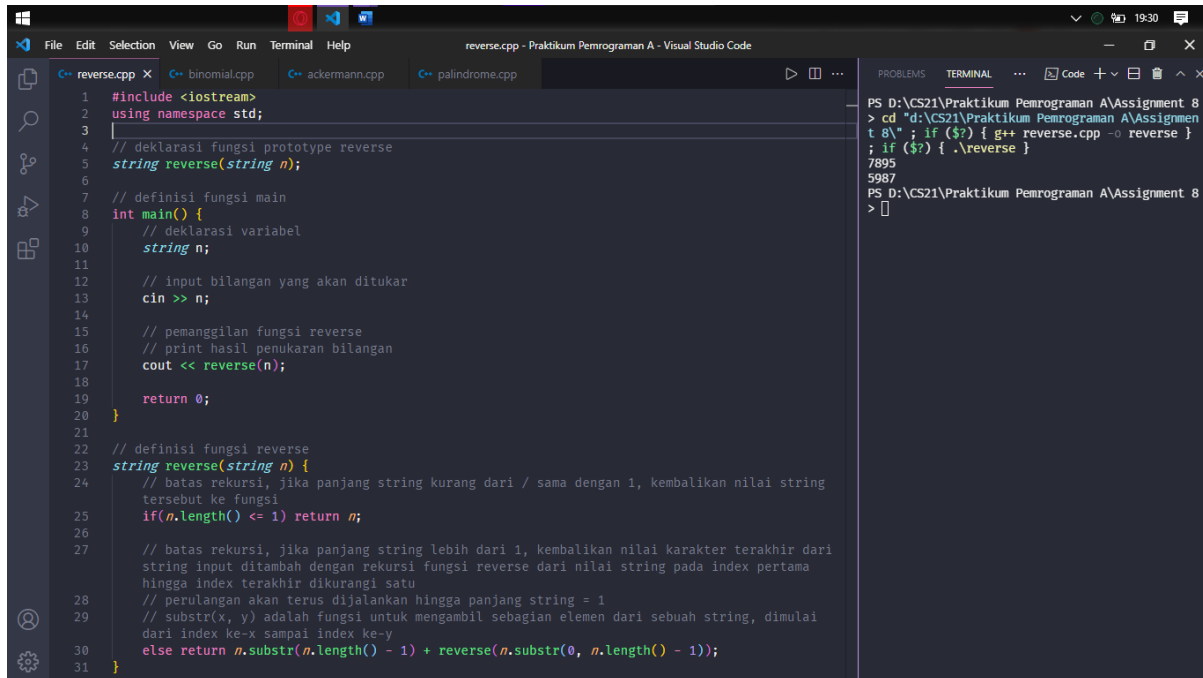
Nama : Andyan Yogawardhana

NIM : 21/482180/PA/21030

Kelas : KOMA

Screenshot

1. Membalik Bilangan

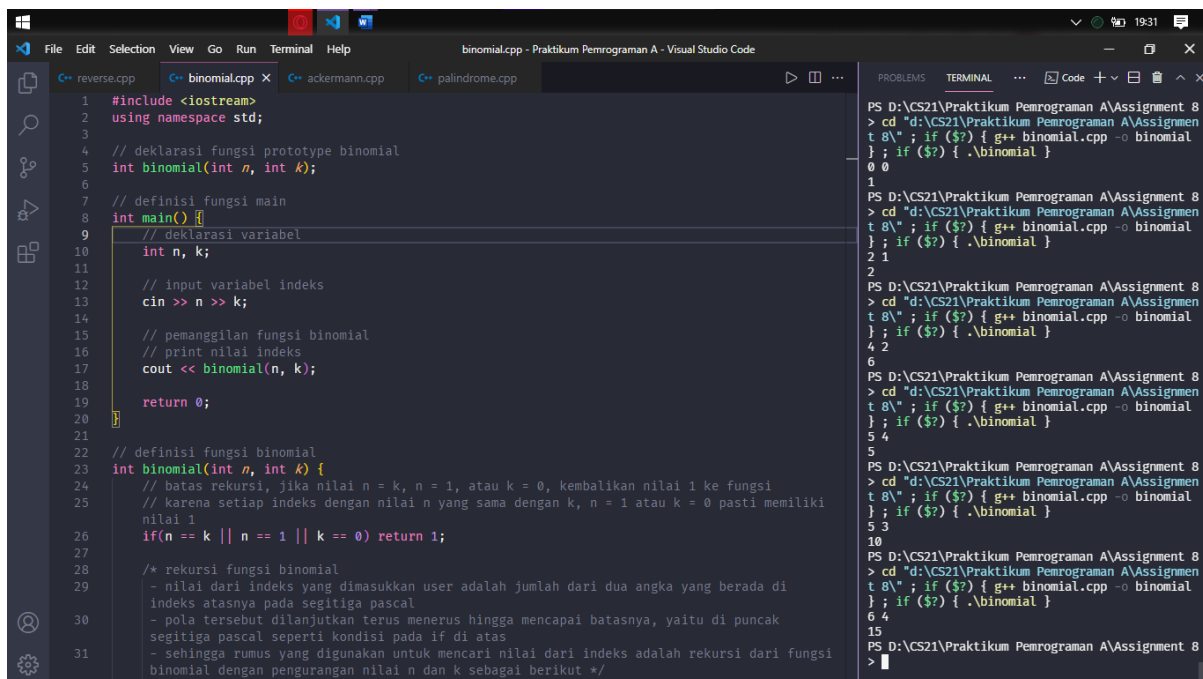


The screenshot shows the Visual Studio Code editor with the file `reverse.cpp` open. The code implements a recursive function to reverse a string. The terminal window on the right shows the command `g++ reverse.cpp -o reverse` and the execution output, which displays the reversed string for the input `87955987`.

```
1 #include <iostream>
2 using namespace std;
3
4 // deklarasi fungsi prototype reverse
5 string reverse(string n);
6
7 // definisi fungsi main
8 int main() {
9     // deklarasi variabel
10    string n;
11
12    // input bilangan yang akan ditukar
13    cin >> n;
14
15    // pemanggilan fungsi reverse
16    // print hasil penukaran bilangan
17    cout << reverse(n);
18
19    return 0;
20 }
21
22 // definisi fungsi reverse
23 string reverse(string n) {
24     // batas rekursi, jika panjang string kurang dari / sama dengan 1, kembalikan nilai string
25     // tersebut ke fungsi
26     if(n.length() <= 1) return n;
27
28     // batas rekursi, jika panjang string lebih dari 1, kembalikan nilai karakter terakhir dari
29     // string input ditambah dengan rekursi fungsi reverse dari nilai string pada index pertama
30     // hingga index terakhir dikurangi satu
31     // perulangan akan terus dijalankan hingga panjang string = 1
32     // substr(x, y) adalah fungsi untuk mengambil sebagian elemen dari sebuah string, dimulai
33     // dari index ke-x sampai index ke-y
34     else return n.substr(n.length() - 1) + reverse(n.substr(0, n.length() - 1));
35 }
```

```
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8"
> g++ reverse.cpp -o reverse
7895
5987
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
>
```

2. Koefisien Binomial



The screenshot shows the Visual Studio Code editor with the file `binomial.cpp` open. The code implements a recursive function to calculate binomial coefficients. The terminal window on the right shows the command `g++ binomial.cpp -o binomial` and the execution output, which displays the binomial coefficients for the input `5 3`.

```
1 #include <iostream>
2 using namespace std;
3
4 // deklarasi fungsi prototype binomial
5 int binomial(int n, int k);
6
7 // definisi fungsi main
8 int main() {
9     // deklarasi variabel
10    int n, k;
11
12    // input variabel indeks
13    cin >> n >> k;
14
15    // pemanggilan fungsi binomial
16    // print nilai indeks
17    cout << binomial(n, k);
18
19    return 0;
20 }
21
22 // definisi fungsi binomial
23 int binomial(int n, int k) {
24     // batas rekursi, jika nilai n = k, n = 1, atau k = 0, kembalikan nilai 1 ke fungsi
25     // karena setiap indeks dengan nilai n yang sama dengan k, n = 1 atau k = 0 pasti memiliki
26     // nilai 1
27     if(n == k || n == 1 || k == 0) return 1;
28
29     /* rekursi fungsi binomial
30     - nilai dari indeks yang dimasukkan user adalah jumlah dari dua angka yang berada di
31     indeks atasnya pada segitiga pascal
32     - pola tersebut dilanjutkan terus menerus hingga mencapai batasnya, yaitu di puncak
33     segitiga pascal seperti kondisi pada if di atas
34     - sehingga rumus yang digunakan untuk mencari nilai dari indeks adalah rekursi dari fungsi
35     binomial dengan pengurangan nilai n dan k sebagai berikut */
36     return binomial(n-1, k) + binomial(n-1, k-1);
37 }
```

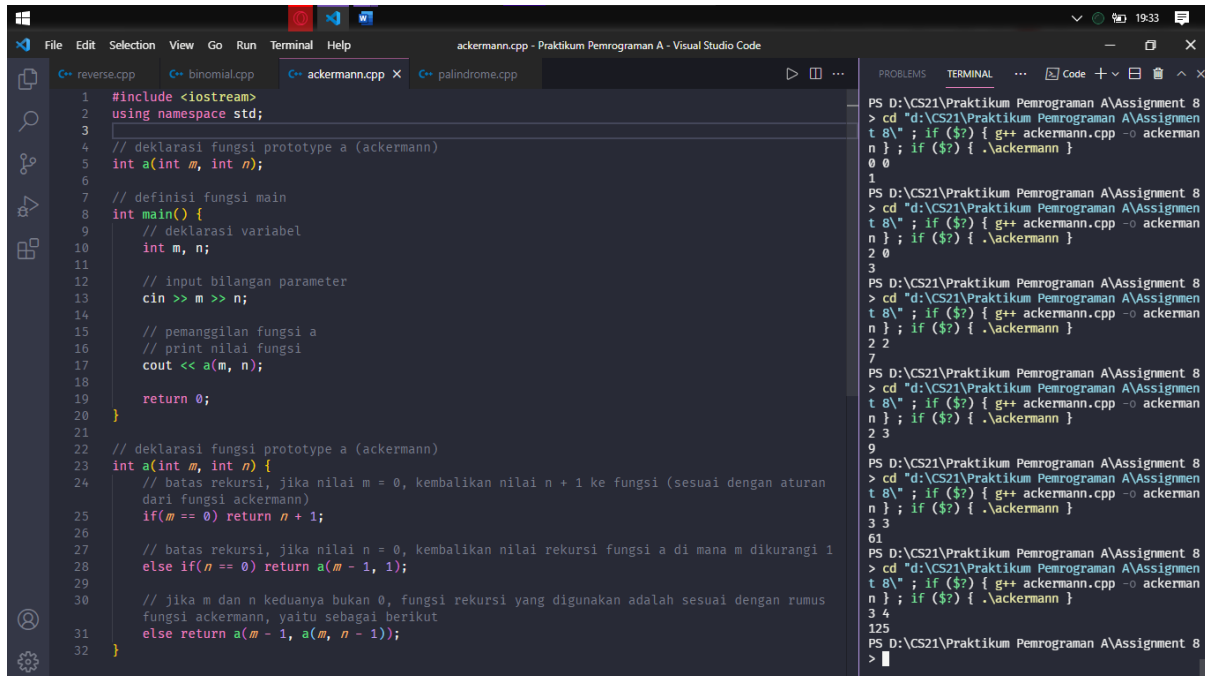
```
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8"
> g++ binomial.cpp -o binomial
5
3
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
>
```

Nama : Andyan Yogawardhana

NIM : 21/482180/PA/21030

Kelas : KOMA

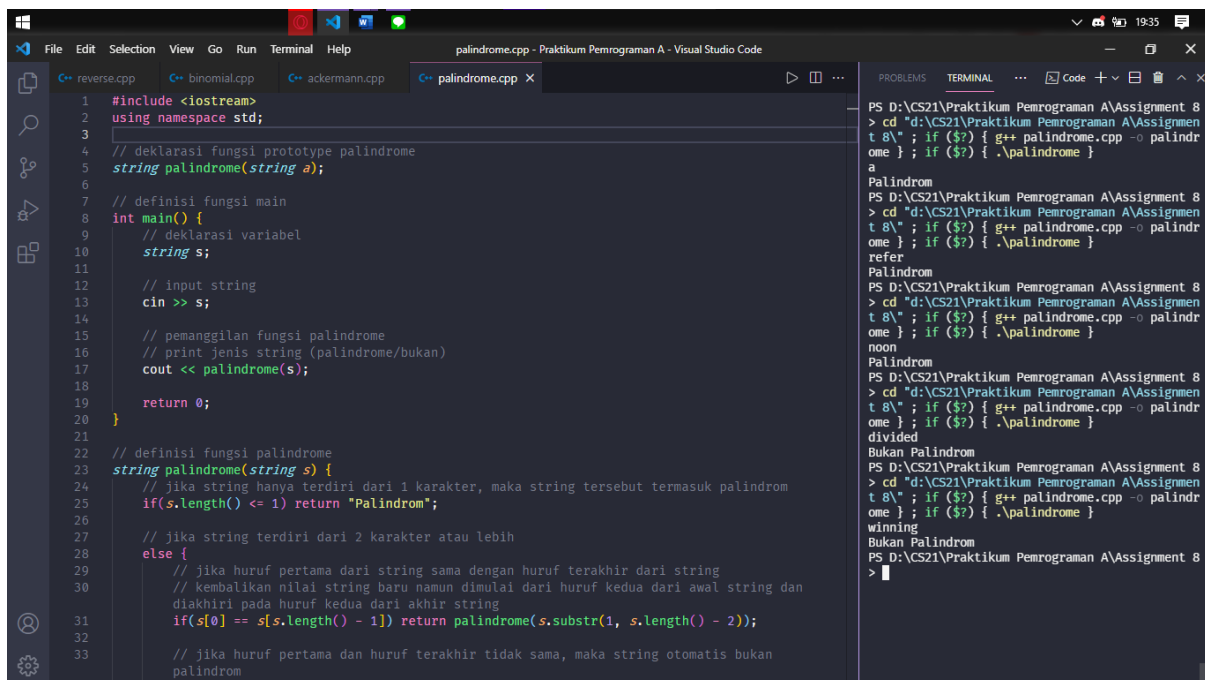
3. Fungsi Ackermann



```
1 #include <iostream>
2 using namespace std;
3
4 // deklarasi fungsi prototype a (ackermann)
5 int a(int m, int n);
6
7 // definisi fungsi main
8 int main() {
9     // deklarasi variabel
10    int m, n;
11
12    // input bilangan parameter
13    cin >> m >> n;
14
15    // pemanggilan fungsi a
16    // print nilai fungsi
17    cout << a(m, n);
18
19    return 0;
20
21 // deklarasi fungsi prototype a (ackermann)
22 int a(int m, int n) {
23     // batas rekursi, jika nilai m = 0, kembalikan nilai n + 1 ke fungsi (sesuai dengan aturan
24     // dari fungsi ackermann)
25     if(m == 0) return n + 1;
26
27     // batas rekursi, jika nilai n = 0, kembalikan nilai rekursi fungsi a di mana m dikurangi 1
28     else if(n == 0) return a(m - 1, 1);
29
30     // jika m dan n keduanya bukan 0, fungsi rekursi yang digunakan adalah sesuai dengan rumus
31     // fungsi ackermann, yaitu sebagai berikut
32     else return a(m - 1, a(m, n - 1));
33 }
```

```
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ ackermann.cpp -o ackermann } & if ($?) { .\ackermann }
0 0
1
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ ackermann.cpp -o ackermann } & if ($?) { .\ackermann }
2 0
3
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ ackermann.cpp -o ackermann } & if ($?) { .\ackermann }
2 2
7
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ ackermann.cpp -o ackermann } & if ($?) { .\ackermann }
2 3
9
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ ackermann.cpp -o ackermann } & if ($?) { .\ackermann }
3 3
61
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ ackermann.cpp -o ackermann } & if ($?) { .\ackermann }
3 4
125
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
>
```

4. Palindrom



```
1 #include <iostream>
2 using namespace std;
3
4 // deklarasi fungsi prototype palindrome
5 string palindrome(string s);
6
7 // definisi fungsi main
8 int main() {
9     // deklarasi variabel
10    string s;
11
12    // input string
13    cin >> s;
14
15    // pemanggilan fungsi palindrome
16    // print jenis string (palindrome/bukan)
17    cout << palindrome(s);
18
19    return 0;
20
21 // definisi fungsi palindrome
22 string palindrome(string s) {
23     // jika string hanya terdiri dari 1 karakter, maka string tersebut termasuk palindrome
24     if(s.length() <= 1) return "Palindrome";
25
26     // jika string terdiri dari 2 karakter atau lebih
27     else {
28         // jika huruf pertama dari string sama dengan huruf terakhir dari string
29         // kembalikan nilai string baru namun dimulai dari huruf kedua dari awal string dan
30         // diakhiri pada huruf kedua dari akhir string
31         if(s[0] == s[s.length() - 1]) return palindrome(s.substr(1, s.length() - 2));
32
33         // jika huruf pertama dan huruf terakhir tidak sama, maka string otomatis bukan
34         // palindrome
35         else return "Bukan Palindrome";
36     }
37 }
```

```
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ palindrome.cpp -o palindrome } & if ($?) { .\palindrome }
a
Palindrome
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ palindrome.cpp -o palindrome } & if ($?) { .\palindrome }
refer
Palindrome
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ palindrome.cpp -o palindrome } & if ($?) { .\palindrome }
noon
Palindrome
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ palindrome.cpp -o palindrome } & if ($?) { .\palindrome }
divided
Bukan Palindrome
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
> cd "d:\CS21\Praktikum Pemrograman A\Assignment 8" & if ($?) { g++ palindrome.cpp -o palindrome } & if ($?) { .\palindrome }
winning
Bukan Palindrome
PS D:\CS21\Praktikum Pemrograman A\Assignment 8
>
```