

Tugas Week 7 : Bahasa Malang

Qornain Aji

21/481767/TK/53170

Pada tugas kali ini, kami diperintahkan untuk membuat algoritma yang dapat mengubah kata/kalimat yang normal menjadi kata/kalimat dengan bahasa slang Malang. Sepengetahuan saya, ketika terdapat kata/kalimat seperti "malang", maka kata tersebut akan diubah menjadi "ngalam". Perhatikan bahwa semua huruf-hurufnya memiliki urutan yang terbalik, kecuali huruf "ng". Adapun kalimat seperti "tangan malang" maka akan diubah menjadi "nangat ngalam". Perhatikan bahwa tata letak urutan kata tidak berubah sehingga tetap bisa dibaca secara SPOK. Untuk itu kami membuat algoritma yang dapat mengubah urutan huruf-hurufnya tanpa menghilangkan urutan SPOK nya dengan code source sebagai berikut.

```
//Qornain Aji
//21/481767/TK/53170

#include <iostream>
#include <stack>
#include <string>
#include <conio.h>
using namespace std;

int main()
{
    stack<char> malang;
    string kata;

    char x;
    do{
        cout << "*** My Lovely Malang Word <3 ***" << endl;
        cout << "    Pilih menu: " << endl;
        cout << "    1. Tambahkan tulisan" << endl;
        cout << "    2. Konversi ke Bahasa Malang" << endl;
        cout << "    3. Exit (Press any number)" << endl << endl;

        cout<<"\nMasukkan angka: ";
        x = getch();
        switch (x)
        {
            case '1':
                cout<<"\nMasukkan kalimat : ";
```

```

        getline(cin,kata);
        break;
    case '2':
        cout<<"Bahasa Malang : ";
        for(int i=0; i<kata.length(); i++)
        {
            if(kata[i]=='n' && kata[i+1]=='g')
            {
                kata[i]='g';          //mengubah urutan "ng" menjadi "gn"
                kata[i+1]='n';        //sehingga ketika dimasukkan ke stack
            }                          //akan tetap keluar menjadi "ng"
        }
        for(int i=0; i<kata.length(); i++)
        {
            if(kata[i]!=' ')
            {
                malang.push(kata[i]); //ketika tidak menemukan spasi,
            }                          //kata dimasukkan ke dalam stack
            else if(kata[i]==' ')
            {
                while(!malang.empty()) //ketika menemukan spasi
                {                       //stack akan di display
                    cout<<malang.top(); //dan akan dioutputkan spasi
                    malang.pop();       //dan setelah stack kosong, stack
                                        akan diisi kembali
                }
                cout<<' ';
            }
        }
        for(int i=0; i<kata.length(); i++)
        {
            cout<<malang.top();        //digunakan ketika kalimat tidak ada spasi
            malang.pop();              //di dalam nya
        }
        cout<<"\n\n";
        kata.clear();

        break;
    }

}while((x == '1') || (x == '2'));
system("PAUSE");
getch();
}

```

Output:

```
*** My Lovely Malang Word <3 ***
Pilih menu:
1. Tambahkan tulisan
2. Konversi ke Bahasa Malang
3. Exit (Press any number)

Masukkan angka:
Masukkan kalimat : tangan
```



```
*** My Lovely Malang Word <3 ***
Pilih menu:
1. Tambahkan tulisan
2. Konversi ke Bahasa Malang
3. Exit (Press any number)

Masukkan angka: Bahasa Malang : nangat
```

```
*** My Lovely Malang Word <3 ***
Pilih menu:
1. Tambahkan tulisan
2. Konversi ke Bahasa Malang
3. Exit (Press any number)

Masukkan kalimat : tangan belang
```



```
*** My Lovely Malang Word <3 ***
Pilih menu:
1. Tambahkan tulisan
2. Konversi ke Bahasa Malang
3. Exit (Press any number)

Masukkan angka: Bahasa Malang : nangat ngaleb
```

Setelah hasil serangkaian ujian, bahkan dengan dimasukkan beberapa unit kata, algoritma tersebut berjalan dengan lancar dan memberikan output yang diharapkan.