

**IK141 Struktur Data**

**Struktur Data**

**Stack**



**Di Susun Oleh :**

**Muhammad Rasyid Nurrohim (2208028)**

**PROGRAM STUDI PENDIDIKAN ILMU KOMPUTER**

**FAKULTAS PENDIDIKAN MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS PENDIDIKAN INDONESIA**

**27 Maret 2023**

## 1. Implementasi dan Hasil

### Implementasi dan Hasil

#### Soal no 1

```
1  #include <stdio.h>
2  #include <string.h>
3  #include <stdlib.h>
4  #define MAX 1000
5
6  typedef struct{
7      int data[MAX];
8      int top;
9  }Stack;
10
11 void push(Stack *s, char n){
12     s->top++;
13     s->data[s->top] = n;
14 }
15
16
17 void pop(Stack *s){
18     char n = s->data[s->top];
19     s->top--;
20     return n;
21 }
22
23 int cek(char *str){
24     Stack s;
25     s.top = -1;
26     int i, len = strlen(str);
27
28     for(i = 0; i < len; i++){
29         push(&s, str[i]);
30     }
31     for(i = 0; i < len; i++){
32         if(str[i] != pop(&s)){
33             return 0;
34         }
35     }
36     return 1;
37 }
```

```

37
38 int main(){
39     char str[MAX];
40     scanf("%s", str);
41     int len = strlen(str);
42
43     printf("\n %d \n", len);
44     if(check(str)){
45         printf("polindrom");
46     }else{
47         printf("bukna polindrom");
48     }
49 }

```

```

malam
5
palindrom
-----
Process exited
Press any key

```

```

C:\Users\ACER\Desktop
matahari
8
bukan palindrom
-----
Process exited after
Press any key to c

```

## Soal no 2

```

1  #include <stdio.h>
2  #include <math.h>
3  #include <string.h>
4  #define MAX 1000
5
6  typedef struct {
7      int top;
8      int data[MAX];
9  } Stack;
10
11 void push(Stack *s, int n) {
12     s->top++;
13     s->data[s->top] = n;
14 }
15
16 int pop(Stack *s) {
17     int n = s->data[s->top];
18     s->top--;
19     return n;
20 }
21

```

```

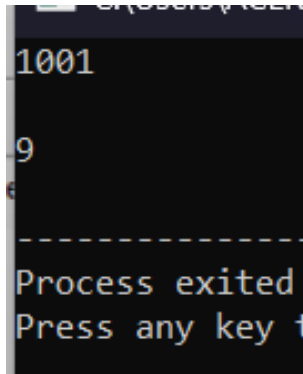
21
22 int ubah(char *binner) {
23     Stack s;
24     s.top = -1;
25     int i, len = strlen(binner);
26     int a = 0;
27     int bit;
28     for (i = 0; i < len; i++) {
29         bit = binner[i] - '0';
30         push(&s, bit);
31     }
32     for (i = 0; i < len; i++) {
33         bit = pop(&s);
34         a += bit * pow(2, i);
35     }
36     return a;
37 }
38

```

```

38
39 int main() {
40     char binner[MAX];
41     int bit;
42     scanf("%s", binner);
43     bit = ubah(binner);
44     printf("\n%d\n", bit);
45     return 0;
46 }
47

```



### Soal 3

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #define MAX 100
5
6 typedef struct {
7     char ket[50];
8     int jumlah;
9 } TransaksiKeuangan;
10
11 typedef struct {
12     TransaksiKeuangan transaksi[MAX];
13     int top;
14 } Stack;
15
16 void push(Stack *s, char *ket, int jumlah) {
17     if (s->top == MAX-1) {
18         printf("Stack penuh\n");
19     } else {
20         s->top++;
21         strcpy(s->transaksi[s->top].ket, ket);
22         s->transaksi[s->top].jumlah = jumlah;
23     }
24 }

```

```

25 void pop(Stack *s) {
26     if (s->top == -1) {
27         printf("Stack kosong\n");
28     } else {
29         printf("Transaksi terakhir telah dihapus\n");
30         s->top--;
31     }
32 }
33
34
35 void display(Stack s) {
36     if (s.top == -1) {
37         printf("Daftar transaksi kosong\n");
38     } else {
39         printf("Daftar transaksi keuangan:\n");
40         for (int i = s.top; i >= 0; i--) {
41             printf("%s - Rp %d\n", s.transaksi[i].ket, s.transaksi[i].jumlah);
42         }
43     }
44 }

```

```

46 int main() {
47     Stack s;
48     s.top = -1;
49     int pil, jumlah;
50     char ket[50];
51     do {
52         printf("\n===== DAFTAR TRANSAKSI KEUANGAN =====\n");
53         printf("1. Tambah transaksi\n");
54         printf("2. Hapus transaksi terakhir\n");
55         printf("3. Tampilkan daftar transaksi\n");
56         printf("4. Keluar\n");
57         scanf("%d", &pil);
58
59         switch (pil) {
60             case 1:
61                 scanf("%s", ket);
62                 scanf("%d", &jumlah);
63                 push(&s, ket, jumlah);
64                 printf("Transaksi berhasil ditambahkan\n");
65                 break;
66             case 2:
67                 pop(&s);
68                 break;
69
70             case 3:
71                 display(s);
72                 break;
73             case 4:
74                 printf("Keluar dari program\n");
75                 break;
76             default:
77                 printf("pil tidak valid\n");
78                 break;
79         }
80     } while (pil != 4);
81     return 0;
82 }
83

```

```

===== DAFTAR TRANSAKSI KEUANGAN =====
1. Tambah transaksi
2. Hapus transaksi terakhir
3. Tampilkan daftar transaksi
4. Keluar
1
baju
100000
Transaksi berhasil ditambahkan

===== DAFTAR TRANSAKSI KEUANGAN =====
1. Tambah transaksi
2. Hapus transaksi terakhir
3. Tampilkan daftar transaksi
4. Keluar
1
celana
200000
Transaksi berhasil ditambahkan

===== DAFTAR TRANSAKSI KEUANGAN =====
1. Tambah transaksi
2. Hapus transaksi terakhir
3. Tampilkan daftar transaksi
4. Keluar
1
topi
50000
Transaksi berhasil ditambahkan
Warnings: 0

```

```

===== DAFTAR TRANSAKSI KEUANGAN =====
1. Tambah transaksi
2. Hapus transaksi terakhir
3. Tampilkan daftar transaksi
4. Keluar
3
Daftar transaksi keuangan:
topi - Rp 50000
celana - Rp 200000
baju - Rp 100000

===== DAFTAR TRANSAKSI KEUANGAN =====
1. Tambah transaksi
2. Hapus transaksi terakhir
3. Tampilkan daftar transaksi
4. Keluar
2
Transaksi terakhir telah dihapus

===== DAFTAR TRANSAKSI KEUANGAN =====
1. Tambah transaksi
2. Hapus transaksi terakhir
3. Tampilkan daftar transaksi
4. Keluar
3
Daftar transaksi keuangan:
celana - Rp 200000
baju - Rp 100000

```

#### Soal 4

```

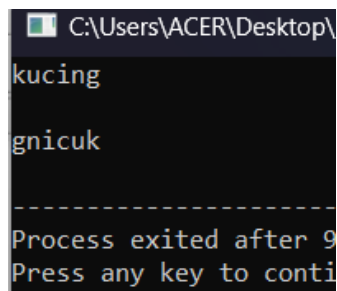
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #define MAX 100
5
6  typedef struct{
7      char data[MAX];
8      int top;
9  } Stack;
10
11 void push(Stack *s, char c) {
12     if (s->top == MAX-1) {
13         printf("Stack penuh\n");
14     } else {
15         s->top++;
16         s->data[s->top] = c;
17     }
18 }

```

```

19
20 char pop(Stack *s) {
21     if (s->top == -1) {
22         printf("Stack kosong\n");
23         return '\0';
24     } else {
25         char c = s->data[s->top];
26         s->top--;
27         return c;
28     }
29 }
30
31 void balikStr(char *str) {
32     Stack s;
33     s.top = -1;
34     for (int i = 0; i < strlen(str); i++) {
35         push(&s, str[i]);
36     }
37     for (int i = 0; i < strlen(str); i++) {
38         str[i] = pop(&s);
39     }
40 }
41
42 void balikStr(char *str) {
43     Stack s;
44     s.top = -1;
45     for (int i = 0; i < strlen(str); i++) {
46         push(&s, str[i]);
47     }
48     for (int i = 0; i < strlen(str); i++) {
49         str[i] = pop(&s);
50     }
51 }
52
53 int main() {
54     char str[MAX];
55     scanf("%s", str);
56     balikStr(str);
57     printf("\n%s\n", str);
58     return 0;
59 }

```



```

C:\Users\ACER\Desktop>
kucing
gnicuk
-----
Process exited after 9
Press any key to conti

```

## 2. Kesimpulan

### Kesimpulan

Pada peraktikum ini saya belajar lebih dalam mengenai stack, mulai dari mengimplementasikan palindrom tdengan stack dan bilangan biner. Dalam stuck juga kita bisa terbantu dalam menyelesaikan masalah. Terdapat beberapa implementasi dasar seperti push, pop, isEmpty dan isFull. Lalu stack juga dapat diimplementasikan dengan menggunakan data struct dan array