Laporan UTS Praktikum SDA

Kelompok 6

Anggota Kelompok:

- Abdullah Asy-Syifawi 2408107010042
- M. Anis Fathin 2408107010045

Program ini menggunakan stack untuk memproses ekspresi aritmatika dalam bentuk infix, postfix, dan prefix. Dalam program ini terdapat 18 fungsi yang di buat, 5 di antaranya berada di file stack.h dan 13 sisanya berada di file passing_expression.c. Adapun file func.h berisi fungsi getch() yang tersedia di library <conio.h> yang awalnya hanya support di OS berbasis Windows menjadi support juga di Linux/Mac/Unix.

Penjelasan Fungsi:

1. clearScr():

- Fungsi ini digunakan untuk membersihkan layar terminal atau console.
- Pada sistem Windows, fungsi ini memanggil perintah system("cls"), sedangkan pada sistem Unix/Linux, ia memanggil perintah system("clear"). Dengan cara ini, program dapat bekerja pada kedua platform tersebut dengan cara yang sesuai.

2. printTitle(char *str):

- Fungsi ini mencetak judul (title) dengan format tertentu pada layar, yang berguna untuk memberikan identifikasi pada setiap menu.
- Menambahkan tanda = di atas dan bawah untuk memberi tampilan yang lebih terstruktur dan menarik, serta memberikan penekanan pada teks yang diproses dengan **bold** (menggunakan \e[1m).

3. menuSelect(char **menu, int len):

- Fungsi ini menampilkan daftar pilihan menu pada layar dan memungkinkan pengguna untuk memilih menu menggunakan tombol w/s atau tombol panah atas/bawah.
- Fungsi ini menggunakan metode pemilihan berbasis keyboard, dan pengguna bisa memilih menu dengan menekan tombol space atau enter. Jika tombol backspace atau esc ditekan, program akan kembali ke menu sebelumnya.

4. expressionCheck(char *str, char *type):

- Fungsi ini digunakan untuk memeriksa apakah ekspresi matematika yang dimasukkan valid berdasarkan jenis ekspresi (Infix, Postfix, atau Prefix).
- Fungsi ini memeriksa posisi operator pertama dan terakhir dalam ekspresi untuk menentukan apakah urutannya benar atau tidak (misalnya, apakah ekspresi dimulai atau diakhiri dengan operator yang tidak sesuai).

5. convertMenu():

- Fungsi ini menangani logika untuk memilih jenis konversi ekspresi (misalnya, Infix ke Postfix, Postfix ke Infix, dll.).
- Menampilkan menu konversi dan meminta pengguna untuk memilih konversi yang diinginkan.
- Setelah memilih, program meminta ekspresi yang akan dikonversi dan melakukan konversi sesuai dengan pilihan menggunakan fungsi-fungsi seperti infixToPostfix, postfixToInfix, dll.
- Setelah konversi selesai, hasilnya ditampilkan di layar.

6. reverse(char* str):

- Fungsi ini membalikkan urutan karakter dalam string.
- Berguna untuk mengubah urutan ekspresi matematika, seperti yang diperlukan dalam konversi dari infix ke prefix dan sebaliknya.

7. precedence(char c):

- Fungsi ini mengembalikan tingkat prioritas (precedence) dari operator yang diberikan.
- Operator yang lebih kuat, seperti ^, memiliki tingkat prioritas lebih tinggi dibandingkan dengan operator *, /, +, dan -.

8. infixToPostfix(char *infix, char *postfix):

- Fungsi ini mengonversi ekspresi dalam notasi infix (yang umum digunakan) menjadi notasi postfix (notasi Polish terbalik).
- Menggunakan stack untuk menyimpan operator dan memastikan operator ditempatkan dengan benar dalam urutan postfix.
- Setelah mengonversi seluruh ekspresi, hasilnya disalin ke dalam variabel postfix.

9. postfixToInfix(char *postfix, char *infix):

- Fungsi ini mengonversi ekspresi dalam notasi postfix menjadi notasi infix.
- Menggunakan stack untuk membangun ekspresi infix dari operand dan operator yang ada di postfix.

10. infixToPrefix(char *infix, char *prefix):

- Fungsi ini mengonversi ekspresi dalam notasi infix menjadi notasi prefix (notasi Polandia).
- Langkah pertama adalah membalikkan ekspresi infix menggunakan fungsi reverse().
- Setelah itu, notasi infix yang dibalik dikonversi menjadi postfix menggunakan infixToPostfix(), dan akhirnya hasil postfix dibalik kembali menjadi prefix.

11. prefixToInfix(char *prefix, char *infix):

- Fungsi ini mengonversi ekspresi dalam notasi prefix menjadi notasi infix.
- Dimulai dengan membalikkan ekspresi prefix, lalu mengonversinya menjadi infix menggunakan stack, dan akhirnya hasilnya dibalik lagi untuk mendapatkan infix yang benar.

12. prefixToPostfix(char *prefix, char *postfix):

- Fungsi ini mengonversi ekspresi dalam notasi prefix menjadi notasi postfix.
- Langkah pertama adalah mengonversi ekspresi prefix menjadi infix menggunakan prefixToInfix(), kemudian infix tersebut dikonversi menjadi postfix dengan infixToPostfix().

13. postfixToPrefix(char *postfix, char *prefix):

- Fungsi ini mengonversi ekspresi dalam notasi postfix menjadi notasi prefix.
- Pertama-tama, ekspresi postfix dikonversi menjadi infix menggunakan postfixToInfix(), kemudian infix tersebut dikonversi menjadi prefix dengan infixToPrefix().

14. initStack(struct stack *stc)

- Fungsi untuk menginisialisasi stack dengan mengatur top menjadi -1, yang berarti stack kosong.

15. isEmpty(struct stack *stc)

- Fungsi untuk mengecek apakah stack kosong.
- Mengembalikan 1 jika stack kosong (top == -1), dan 0 jika tidak.

16. isFull(struct stack *stc)

- Fungsi untuk mengecek apakah stack penuh.
- Mengembalikan 1 jika top == max - 1, artinya stack sudah mencapai kapasitas maksimum.

17. push(struct stack *stc, char *str)

- Fungsi untuk menambahkan elemen (string) ke dalam stack.
- Jika stack belum penuh, string akan disalin ke posisi top + 1, lalu top dinaikkan.
- Jika stack penuh, akan mencetak pesan "Stack penuh...\nPush gagal\n".

18. pop(struct stack *stc, char *str)

- Fungsi untuk menghapus elemen dari stack dan menyimpannya ke variabel str.
- Jika stack tidak kosong, elemen pada posisi top akan disalin ke str, lalu top dikurangi.
- Jika stack kosong, akan mencetak pesan "Stack kosong...\nPop gagal\n".

19. main():

- Fungsi utama yang mengatur alur utama program.
- Menampilkan menu utama dengan pilihan seperti "Mulai", "About", "How to Start", dan "Exit".
- Jika pilihan "Mulai" dipilih, maka akan memanggil convertMenu() untuk memilih jenis konversi ekspresi.
- Jika pilihan "About" dipilih, menampilkan penjelasan mengenai notasi infix, prefix, dan postfix.
- Jika pilihan "How to Start" dipilih, menampilkan petunjuk penggunaan program.