Nama : Li Wei

NIM : 825220054

Jurusan : Sistem Informasi

Mata kuliah : Algoritma dan Programming/Kelas B

1b. Buatlah fungsi rekursif Mencari hasil a kali n (a * n) dengan menggunakan penjumlahan.

Fungsi_kaliperjumlahan

Fungsi ini merupakan fungsi rekursif dimana untuk mencari hasil kali dengan perjumlahan, dimana variable a dan n bertipe integer.

```
int kaliperjumlahan(int a, int n){
 if (n == 0)
  return 0;
 else
  return x + kalijumlah(a, n-1);
}
```

Algoritma utama

1. [mulai]

[Masukkan nilai a]

Read (a)

2. [Masukkan nilai n)

Read (n)

3. [Memanggil fungsinya dan tampilkan hasilnya]

Write (“Perkalian dari” a “dengan” n “adalah” kaliperjumlahan(a,n)

4. [selesai]

Halt

2b. membuat notasi postfix dari (A*B + C/D + E ^ (E div C) -(D -E /(F + G)))

= AB*+ CD/ + E ^ Ediv C DE- F+ G / -

Table proses pengubahan ke postfix

1	A	(	A

2	*	(*	A
3	B	(*	AB
4		(	AB*
5	+	(+	AB*
6		(	AB*+
7	C	(	AB*+ C
8	/	(/	AB*+ C
9	D	(/	AB*+ CD
10		(	AB*+ CD/
11	+	(+	AB*+ CD/
12		(	AB*+ CD/ +
13	E	(	AB*+ CD/ + E
14		(	AB*+ CD/ + E
15	^	(^	AB*+ CD/ + E
16		(	AB*+ CD/ + E ^
17	(	((	AB*+ CD/ + E ^
18	E	((	AB*+ CD/ + E ^ E
19		((	AB*+ CD/ + E ^ E
20	d	((	AB*+ CD/ + E ^ Ed
21	i	((	AB*+ CD/ + E ^ Edi
22	v	((	AB*+ CD/ + E ^ Ediv
23		((	AB*+ CD/ + E ^ Ediv
24	C	((	AB*+ CD/ + E ^ Ediv C
25	)	(	AB*+ CD/ + E ^ Ediv C
26		(	AB*+ CD/ + E ^ Ediv C
27	-	(-	AB*+ CD/ + E ^ Ediv C
28	(	(-(	AB*+ CD/ + E ^ Ediv C
29	D	(-(	AB*+ CD/ + E ^ Ediv C D

30		(-(	AB*+ CD/ + E ^ Ediv C D
31	-	(-(-	AB*+ CD/ + E ^ Ediv C D
32	E	(-(-	AB*+ CD/ + E ^ Ediv C DE
33		(-(	AB*+ CD/ + E ^ Ediv C DE-
34	/	(-(/	AB*+ CD/ + E ^ Ediv C DE-
35	(	(-(/(	AB*+ CD/ + E ^ Ediv C DE-
36	F	(-(/(	AB*+ CD/ + E ^ Ediv C DE- F
37		(-(/(	AB*+ CD/ + E ^ Ediv C DE- F
38	+	(-(/(+	AB*+ CD/ + E ^ Ediv C DE- F
39		(-(/(	AB*+ CD/ + E ^ Ediv C DE- F+
40	G	(-(/(	AB*+ CD/ + E ^ Ediv C DE- F+ G
41	)	(-(/	AB*+ CD/ + E ^ Ediv C DE- F+ G
42	)	(-	AB*+ CD/ + E ^ Ediv C DE- F+ G /
43	)		AB*+ CD/ + E ^ Ediv C DE- F+ G / -
44			AB*+ CD/ + E ^ Ediv C DE- F+ G / -

3b. Setiap mahasiswa mengerjakan soal ujian yang terdiri dari 10 pertanyaan yang memilih satu dariempat pilihan (PG).

i.
```
struct mahasiswa{
        int kode_mahasiswa[100];
        int jawaban[100];
        int nilai[100];
};
```

ii.
```
1. [looping untuk meminta jawaban]
for (int i=1; i<=10; i++){
2. Write  ("Jawaban - " << i << " = ")
  Read  (mhs.jawaban[i])
 }
```

iii.
```
1. [Looping untuk menginput kunci jawaban]
for (int i=1; i<=n; i++){
2. [meminta user untuk menginput kode]
Read (mhs.kode_mahasiswa[i])
3. [Looping untuk menginput kunci jawaban]
for (int j=1; i<=10; i++){
```

```
4. [Menampilkan jawabannya]
Write ("Jawaban - "  j  " = ")
[Meminta user untuk memasukkan nilai]
Read (nilai[j])
        }

}
```

iv.
```
for (int i=1; i<=10; i++){
for (int j=1; i<=10; i++){
if (mhs.jawaban[i] == mhs.nilai[j]){
skor = 10;
}
else {
}
}
```

v.
```
ofstream Ofile ("TI.txt");
for (int i = 0; i < 10; i++) {
if (mhs.kode[i] >= 825220000) {
Ofile << mhs.kode_mahasiswa[i] << jawaban[1] << jawaban[2] << jawaban[3] << jawaban[4]
<< jawaban[5] << jawaban[6] << jawaban[7] << jawaban[8] << jawaban[9] << jawaban[10] <<
skor << endl;
}
}
Ofile.close();

ofstream File ("SI.txt");
for (int i = 0; i < 10; i++) {
if (mhs.kode[i] <= 825220000 && mhs.kode[i] >= 525220000) {
File << mhs.kode_mahasiswa[i] << jawaban[1] << jawaban[2] << jawaban[3] << jawaban[4]
<< jawaban[5] << jawaban[6] << jawaban[7] << jawaban[8] << jawaban[9] << jawaban[10] <<
skor << endl;
}
}
File.close();
```

4b. Kasus soal linked list:

i.
```
#include <bits/stdc++.h>
using namespace std;

struct mahasiswa{
        int kode_mahasiswa[100];
        int jawaban[100];
        int nilai[100];
   *next;
```

```
};

struct mahasiswa *List;
```

ii.
```
mahasiswa *INSERT_END(mahasiswa *List, int data){
    mahasiswa *N;
    mahasiswa *p;
    N = new mahasiswa;
    N -> nilai = data;
    N -> next = NULL;
    if (List != NULL) {
    p = List;
    while (p -> next != NULL) p = p -> next;
    p -> next = N;
    return (List);
    }
    else
    return (N);
    }
```

iii.
```
a
```

vi.
```
for (int i=1; i<=10; i++){
for (int j=1; i<=10; i++){
if (mhs.jawaban[i] == mhs.nilai[j]){
skor = 10;
}
else {
}
}

ofstream Ofile ("TI.txt");
for (int i = 0; i < 10; i++) {
if (*mhs.kode >= 825220000) {
Ofile << mhs.kode_mahasiswa[i] << jawaban*1 << jawaban*2 << jawaban*3 << jawaban*4
<< jawaban*5 << jawaban*6 << jawaban*7 << jawaban*8 << jawaban*9 << jawaban*10 <<
skor << endl;
}
}
Ofile.close();

ofstream File ("SI.txt");
for (int i = 0; i < 10; i++) {
if (mhs.kode[i] <= 825220000 && mhs.kode[i] >= 525220000) {
File << mhs.kode_mahasiswa[i] << jawaban *1 << jawaban*2 << jawaban*3 << jawaban*4 <<
jawaban*5 << jawaban*6 << jawaban*7 << jawaban*8 << jawaban*9 << jawaban*10 << skor
<< endl;
}
```

```
        }
        File.close();
```
iv.