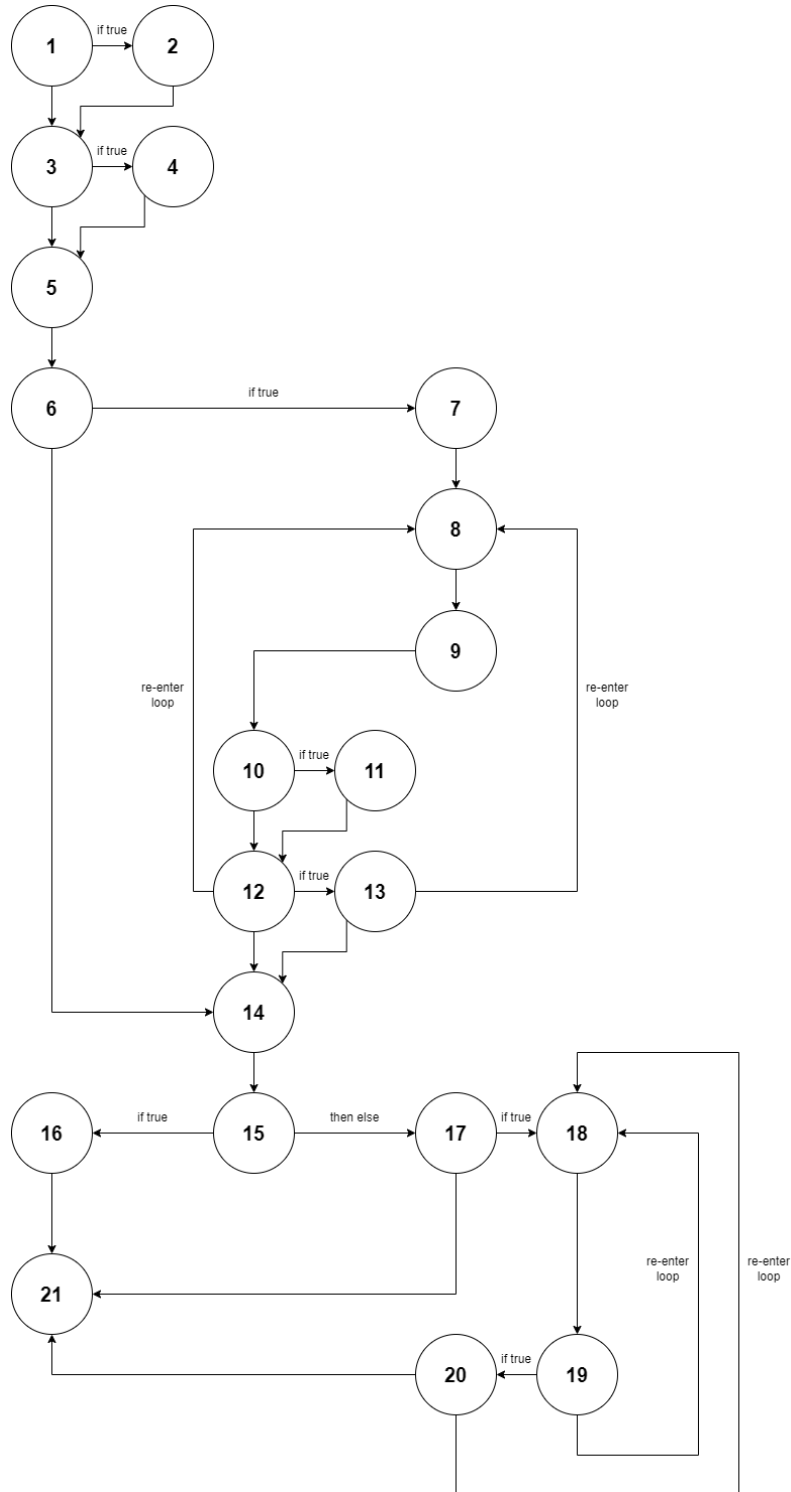


2. Control Flow Graph



3. Цикломатска комплексност

$$M=E-N+2P \dots M=31-21+2 \dots M=12$$

Резултатот го добив преку формулата за цикломатска комплексност $M=E-N+2P$, бидејќи има почетен јазол и завршен кои се различни и не се поврзани во циклус (2 сврзани компоненти). Е го означува бројот на врски, N го означува бројот на јазли и P го означува бројот на сврзани компоненти.

4. Every Branch критериум

Step	A	Б	В	Г
1-2	*		*	
1-3		*		*
2-3	*		*	
3-4	*		*	
3-5		*		*
4-5	*		*	
5-6	*	*	*	*
6-7	*		*	*
6-14		*		
7-8	*		*	*
8-9	*		*	*
9-10	*		*	*
10-11	*		*	
10-12				*
11-12	*		*	
12-8				*
12-13	*		*	
12-14				*
13-8			*	
13-14	*		*	
14-15	*	*	*	*
15-16	*			
15-17		*	*	*
16-21	*			
17-18			*	
17-21		*		*
18-19			*	
19-18			*	
19-20			*	
20-18			*	
20-21			*	

A – Во овој случај патеката поминува по секоја врска каде влегува со исполнет услов.

Започнува со условот (1-2) влегува во него и излегува преку (2-3), потоа исто врските (3-4) и (4-5), продолжува (5-6) па со точен услов низ врската (6-7) и продолжува со (7-8, 8-9, 9-10),

повторно со услов влегува во врска (10-11) и излегува преку (11-12) исто така и за (12-13) и (13-14) избегнувајќи го циклусот во овој случај (постои 1 корисник). Потоа продолжува по врска (14-15) од каде преминува во првиот (точниот) случај од разгранетиот услов со врска (15-16) од каде излегува на крајот преку (16-21).

Б – Во овој случај патеката поминува по секоја врска при што не исполнува услов. Започнува со врска (1-3) при што директно преминува од услов кон услов, потоа (3-5), па понатаму (5-6). Оттука го прескокнува целиот услов и прекинува во следниот јазол преку (6-14). Продолжува со врска (14-15) и во следниот разгранет услов преминува во вториот (неточниот) случај со врска (15-17). Оттаму не влегува во условот што значи излегува на крајот преку (17-21).

В – Во овој случај влегува во условите како во случај **А**, но дополнително се пренасочува во (13-8) што значи поминува еден круг од циклусот (има барем 2 корисници). Се навраќа назад по истите врски но сега од јазол 8 до јазол 15. Оттука влегува во вториот (неточниот) случај на разгранетиот услов преку (15-17). Штом ги следи стапките на случај **А** влегува во наредниот услов (17-18) па (18-19). Го извршува циклусот поголем број пати што значи ја поминува врска (19-18). Продолжува по условот (19-20) па повторно го изминува циклусот но овој пат по врска (20-18) и откако ќе се врати на јазол 20 завршува со врска (20-21).

Г – Овој случај е дополнителен случај на случајот **Б** со тоа што по јазол 6 навидум продолжува како случај **В** и влегува во условот (бидејќи има барем 2 корисници). Ги поминува врските (6-7, 7-8, 8-9, 9-10). Потоа од чекор 10 продолжува по законите на случај **Б** што значи дека продолжува по врските кои не исполнуваат услови, тоа се чекорите (10-12, 12-14), но го користи циклусот преку врска (12-8) (бидејќи има барем 2 корисници). Откако ќе се наврати од чекор 8 до чекор 14 продолжува по врска (14-15) и избегнувајќи го условот преминува во врска (15-17) од каде излегува на крајот со врска (17-21).

На овој начин се поминати сите можни разгранувања.

5. Multiple Condition критериум

Можните комбинации за условот

```
if (user==null || user.getPassword()==null || user.getEmail()==null)
```

се следните:

Постојат 2^3 т.е 8 вкупно комбинации кои треба да се поминат, но се намалуваат на 4 бидејќи 4 од нив не ни се потребни да ги проверуваме.

Во општ случај се следните:

Test Case	user==null	user.getPassword()==null	user.getEmail()==null
1	T	T	T
2	T	T	F

3	T	F	T
4	F	T	T
5	F	F	T
6	F	T	F
7	T	F	F
8	F	F	F

Но, за да убидиме потребни ни се само следните:

Test Case	user==null	user.getPassword()==null	user.getEmail()==null
1	T	X	X
2	F	T	X
3	F	F	T
4	F	F	F

X означува неутрална вредност, т.е не е битно дали е T или F.

За да целосниот услов биде целосно точен несмее да се појави точност во ниту една позиција, што значи единствена точна комбинација е онаа каде сите услови се неточни. Доколку некој услов е точен тоа значи дека тој е всушност празен (не е внесен).